FUNDAMENTALS OF INFORMATION AND COMMUNICATIONS TECHNOLOGY (ICT) FOR CAMEROON SECONDARY SCHOOLS

NKWEAUSEH REGINALD LONGFOR

NKAMENI DANIEL

1st Ed.

Quality Prints, Bamenda, Cameroon.

EDITORS: TCHUENKAM ULRICH
DAISY NKWETEYIM
NGWA PIUS
FOTSO FOKAN EVRARD

Quality Prints Limited

ISBN: 978-9956-15-185-0

Commercial Avenue ,
Mankon, Bamenda,
Cameroon.

FIRST PUBLISHED: 2016

DEDICATION

To God almighty, the creator of humanity and to my lovely parents, ***Lobisi Paul Longfor*** *and* ***Cecilia Kuoh Tufon Longfor****, for they gave me life, nurtured me and helped me to remain healthy.*

- Nkweauseh Reginald Longfor

This book is dedicated to my mother, ***Pastor Mewe Jeannette****, to my junior brother* ***Nkameni Ephraim****, and to all those that have a passion for technology and computer science.*

- Nkameni Daniel

ACKNOWLEDGEMENT

The accomplishment of this work was not a day's job but its eventual completion was made possible by numerous individuals and above all THE divine grace. Our deepest gratitude go to the almighty God who gave us the required inspiration and the spirit of determination and focus to realize this work.

Our greatest thanks and deepest appreciation go to the following people whose contributions have either been technical, financial, moral or otherwise:

Prof. Victor Julius Ngoh, Justice Chibili, Ngueti Michel (Ph.D), Asong Fred Zisuh (Ph.D), Sone Ekonde Michael (Ph.D), Mih Thomas Attia (Ph.D), Tatepong Francois (Ph.D), Nkwemoh Clement (Ph.D), Jumbam Blaise (Msc.), Victoire Ngounoue Epse Ngum (Msc.), Bih Sylvian Tufon, Tewagnoh Harrignton, Nkeumtoh Carelle, Nziembi Laura Longfor, Tisate Lesly Kitu, Piankeh Albert, Ndong Tang Godwin, Emmanuel Mbemoun, Ngwa Franklin, Ngwa Andrew, Ndifor Hyacinth, Megouo Ulrich, Takam Nelino, Mrs Tendongafac A.M, Mr. Ngong Charles, Mr. Ngong Emmanuel, Mr. Che Gilles, Mr Chefuh James M, Mrs Noumbissi Evelyn, Mr. Mikah Edwin, Mr. Okah Alang, Falo Fleury, Mrs Kum Dorothy, Mr Chi Emmanuel.

TABLE OF CONTENT

PREFACE

I have the world's nicest students. They are polite, earnest, and sweet. They are fun to talk to and a delight to teach. Unfortunately, many of them are ill-prepared for college. They are not ready for the onslaught of work. Moreover, many have not been taught neither I.C.T nor computer science both at the Basic education level and First cycle of Secondary Education.

After interviewing students across the country - including many at the nation's top schools - I have come to realize that this dilemma is not unique to my students. Far from it. Worst of all, a distressing number of students believe they cannot succeed. They have been shaken by years of low grades or grade inflation that results in artificially-raised scores.

This book is designed to help *all* students master the *Fundamental concepts of I.C.T* that they need to succeed in their studies. Best of all, when students understand the underpinnings of their specialties (ICT), they turn to grasp the knowledge at the Higher Education Level (College), which inturn prepares them for a better carrier in related domains of specialty such as Information Technology, Computer Science, Software Engineering, Electronic Engineering, and Telecommunication Engineering —as it should be.

- NKWEAUSEH REGINALD LONGFOR, B.Tech(Hons)

G.B.H.S Bafoussam

ORGANISATION OF THE TEXT

This book is arranged in two sections for a total of sixteen chapters. The chapters take you step-by-step through the process of learning the Fundamental concepts in ICT.

Each chapter ends with a series of review exercises (Past CGCE Questions). These help you reinforce and extend what you learned. The exercises include some multiple choice and long structured test items.

Here's how to use this book:

Option 1

● Read through the book from the beginning to the end as you would any book.

● Complete all the exercises at the end of each chapter to assess your progress. This gives you even more practice.

Option 2

● Pick and choose the chapters you wish to read, or read them in any order you like.

● Skim the exercises to find the ones that help you learn more about the areas in which you need improvement.

Option 3

● Use the book as a study guide right before and after major tests. Read and reread the chapters you need the most.

● Complete the exercises that directly match the types of tests you are taking now or plan to take in the immediate future.

Option 4

● Use the internet to compliment your study with this book, and also to gain a better and diverse understanding of each topic covered by this book.

● Section II is highly recommended to be read and the tasks immediately executed on a computer (desktop or laptop). This will permit you to demystify the concepts tested under the practical evaluations in *all* exams. As a result, such exercise will also enable you develop creative and innovative approaches to similar or related tasks.

PART ONE: THEORY

CHAPTER 1: ICT AND COMPUTER SYSTEMS

1.1 INTRODUCTION

What is ICT?

ICT is an acronym that stands for **Information Communications Technology.**

However, apart from explaining an acronym, there's no universally accepted defininition of ICT. *Why?* Because the concepts, methods and applications involved in ICT are constantly evolving on an almost daily basis and it's difficult to keep up.

A good way to think about ICT is to consider all uses of digital technology that exist to help individuals, businesses and organisations use information. ICT covers any product that will store, retrieve, manipulate, transmit or receive information electronically in a digital form. For example, personal computers, digital television, email, robots.

So ICT is concerned with the storage, retrieval, manipulation, transmission or receipt of digital data. Importantly, it is also concerned with the way these different uses can work with each other.

ICT is often classified into two broad categories:

1. ***Traditional computer-based technologies*** (things you can typically do on a personal computer or using computers at home or at work).

2. ***Digital communication technologies*** (which allow people and organisations to communicate and share information digitally).

Let's take a brief look at these two categories to demonstrate the kinds of products and ideas that are covered by ICT:

1- Traditional computer based technologies

a) Standard office applications

i) ***Word processing***: eg MS Word to write letters, reports etc.

ii) ***Spreadsheets***: eg MS Excel to analyse financials, calculations, create forecasting models etc.

iii**)** ***Database software***: eg Oracle/MS SQL Server/Access to manage date in many forms from basic lists (eg customer contacts to catalogues).

iv) ***Presentation software***: eg MS PowerPoint to make presentations.

v) ***Desktop publishing***: eg Adobe Indesign/Quark Express/MS Publisher to produce newsletters, magazines and other complex documents.

vi) ***Graphics software***: eg Adobe Photoshop and Illustrator to create and edit images such as logos, drawing or pictures for use in DTP, websites or other publications.

b) Specialist applications

i) ***Accounting package***: eg Sage/Oracle to manage an organisations accounts.

ii) ***Computer Aided Design (CAD)***: To assist the design process. Specialist programmes exist for many times of design such as architectural, engineering, electronics and roadways.

iii) ***Customer Relations Management (CRM)*****:** to allow businesses to better understand their customers by collecting and analysing data such as their product preferences and buying habits etc. Often linked to software applications that run call centres and loyalty cards, for example.

2- Digital communication technologies

Communication of data by electronic means, usually over some distance is often achieved via networks of sending and receiving equipment, wires and satellite links.

The technologies involved in communication tend to be complex. You certainly don't need to understand them for your ICT course. However, there are aspects of digital communications that you need to be aware of. These relate primarily to the two types of network and the ways of connecting to the internet. Let's look at these two briefly:

a) **Internal networks:** Usually referred to as a *local area network (LAN)*, this involves linking a number of hardware items (input and output devices plus computer processing) together within an office or building.

The aim of a LAN is to be able to share *hardware* facilities such as printers or scanners, software applications and data. This type of network is invaluable in the office environment where colleagues need to have access to common data or programmes.

b) **External networks:** Often you need to communicate with someone outside your internal network, in this case you will need to be part of a wide area network (WAN). The internet is the ultimate WAN - it is a vast network of networks.

In a Broader context, your ICT course will almost certainly cover the above examples of ICT in action, perhaps focusing on the use of key applications such as spreadsheets, databases, presentation, graphics and web design software.

It will also consider the following important topics that deal with the way ICT is used and managed in an organisation and the society at large:

1. ***Nature of information*** *(the "**I**" in **ICT**):* this covers topics such as the meaning and value of information, how information is controlled, the limitations of ICT and legal considerations.

2. ***Management of information***: this covers how data is captured, verified and stored for effective use; the manipulation, processing and distribution of information; keeping information secure and designing networks to share information

3. ***Information systems strategy***: This considers how ICT can be used within a business or organisation as part of achieving goals and objectives.

As you can see, ICT is a broad and fast-changing subject. We hope this book will help you master the concepts in this exciting field - ICT!

1.1.1 ICT SYSTEMS

An ICT system is a set-up consisting of hardware, software, data and the people who use them. It commonly includes communications technology, such as the internet.

ICT and computers are *not* the same thing.

Computers are the hardware that is often part of an ICT system.

This is why it's important not just to learn about computers but about how, why and when people use them. It is the power of computers and communications that has allowed ICT systems to become so important. Like any piece of equipment, the important thing about it is what it lets us do.

Figure 1.1: *Image showing the impact of ICT in life*

ICT systems are used in a number of environments, such as: offices, shops, factories, aircraft, and ships. They're also used in fields such as: communications, medicine, and farming.

ICT systems are every day and ordinary, yet extraordinary in how they can add extra power to what we do and want to do.

1.1.2 The importance of ICT systems

By using ICT systems we are:

- *more productive* - we can complete a greater number of tasks in the same time at reduced cost by using computers than we could prior to their invention
- able to deal with vast amounts of information and process it quickly
- able to transmit and receive information rapidly

The three main types of ICT systems to consider are:

a) Information systems: This type of ICT system is focused on managing data and information. Examples of these are a sports club membership system or a supermarket stock system.

b) **Control systems:** These ICT systems mainly control machines. They use input, process and output, but the output may be moving a robot arm to weld a car chassis rather than information.

c) **Communications systems:** The output of these ICT systems is the successful **transport of data** from one place to another.

1.1.3 Input, output and system diagrams

What comes out of an ICT system is largely dependent on what you put into the system to begin with.

ICT systems work by taking inputs (instructions and data), processing them and producing outputs that are stored or communicated in some way. The higher the quality and better thought-out the inputs, the more useful the outputs.

1.1.4 Garbage In, Garbage Out (GIGO)

ICT systems cannot function properly if the inputs are inaccurate or faulty; they will either not be able to process the data at all, or will output data which is erroneous or useless.

GIGO is a useful term to remember - it can help explain many issues such as why validation is needed and why accurate data is valuable.

1.1.5 An ICT system diagram

A system is an assembly of parts that together make a whole. ICT systems are made up of some or all of the parts shown in the diagram. Various devices are used for input, processing, output, and communication.

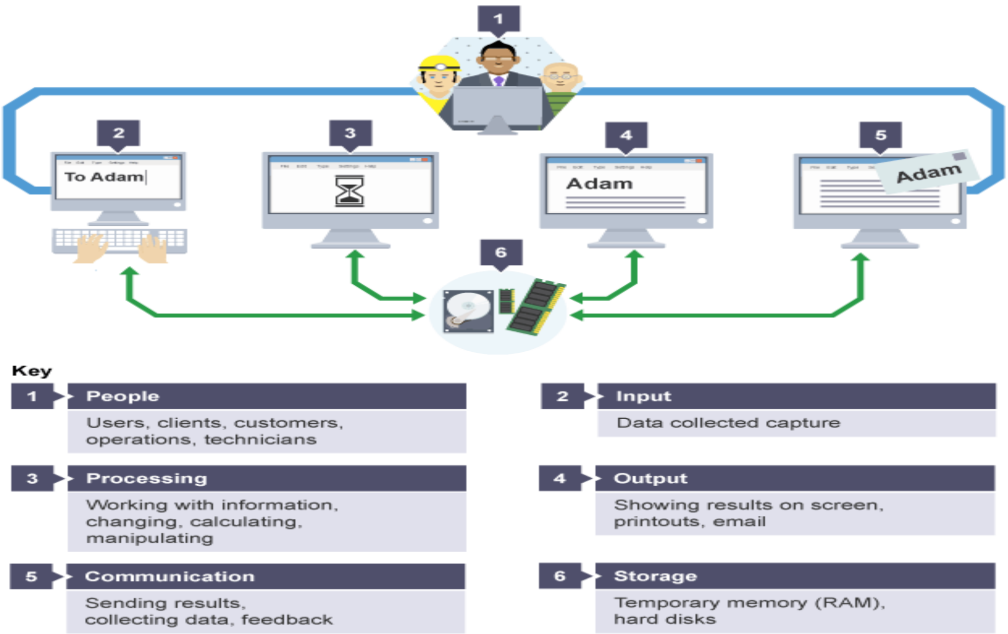


Figure 1.2: *ICT system diagram*

1.1.6 Feedback

It is sometimes good to have feedback in an ICT system. This is when the output from a system **feeds back** to influence the input and the process repeats itself.

A good example is a system set-up to control water temperature in a tropical fish tank. The temperature of the water is taken as an input from sensors. Processing takes place and the temperature of the water is compared against the pre-programmed parameters, eg maximum/minimum temperature. The outputs include the automatic decision to either turn on or off the heater to warm or let the water cool. The output, i.e the change in the water's temperature, is then fed back by the sensors as an input and the process repeats itself.

Feedback can occur in information-based systems as well. Often an output will have a result on further inputs. For example, the output of accepting an online booking for an air ticket will be to reduce the number of tickets available.

1.1.7 Media integration

Methods used for input to and output from ICT systems vary a lot. Input and output formats are the different kinds of media that are used to either gather up and collect data and instructions or to display, present or issue the outputs of processing.

Up until recently most media formats required dedicated devices - for example, digital cameras to take digital photographs, scanners to digitize images for use on a computer, or DVD players for video playback - so you needed the correct device in order to work with each media format.

There is now a growing tendency for multi-purpose ICT devices, which is known as conversion. The driving force is the communication power of the internet, and the increasing availability of small high-powered electronic technology. This means that you can now get an all-in-one box that can do the same thing as several different ones did before it. Here are some examples:

Figure 1.3: *Media Integration*

- Combined printers, scanners and photocopiers.
- Televisions with built-in internet connections and web browsers.
- Mobile phones that can take photos, record video, access the internet and play back music.
- Applications that allow phones to do even more things beyond taking photos, videos or running browsers. For example, the phone all-in-one with a camera allows you to take pictures, but an app will allow you to edit it, add a filter and send it overseas for free.

1.1.8 Integration of information services

Alongside the joining together of technologies, there is also a tendency toward the integration of common public information services.

Digital television by satellite, cable or terrestrial aerial now gives access to many channels that have interactive content, which can be used in a similar way to the web. DAB digital radio provides large amounts of text data to be transmitted along with the signal. The internet enables broadcasts from radio and TV stations to be 'time-shifted' by the user, who watches or listens to the programme whenever they want to.

Figure 1.4: *integration of information services*

News services and the mass media such as newspapers, radio, and television are making themselves available so that people can access them when they want and wherever they are. Two of the key reasons for this are:

1. the growth of broadband internet access that allows lots of information to be viewed quickly and effectively
2. the success of digital broadcasting, the signals of which can carry very much more content than old style analogue TV and radio

1.2 COMPUTER SYSTEMS:

A computer system is one that is able to take a set of inputs, process them and create a set of outputs. This is done by a combination of hardware and software

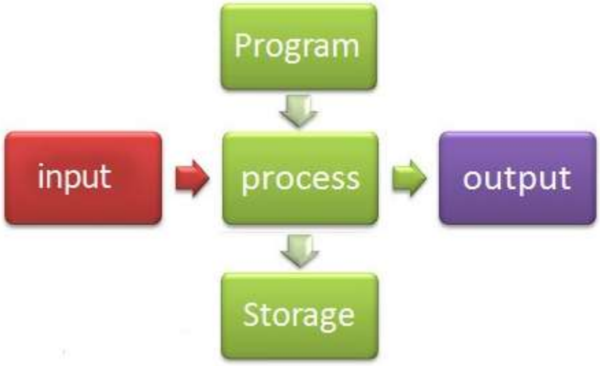


Figure 1.5: *functional block diagram of a computer system*

The computer system has one or more **inputs** to provide data. This data is then **processed** in some way. The outcome of the processing is sent to an **output** or it may be **stored** until some event happens to cause it to be output. For processing to take place, there needs to be a set of instructions of what needs to be done. This set of instructions is called a *program*. This system is called a *stored-program computer*.

The very first commercial computer in the entire world of this type was built in Britain in 1949 and was called the Manchester (Ferranti) Mark 1, see beside. The beauty of this type of computer system is that it is flexible - the machine performs a different task by simply loading a different program from storage.

Figure 1.6: *first commercial computer*

1.2.1 TYPES OF COMPUTERS:

Computers can be generally classified by size and power as follows, though there is considerable overlap:

- **Personal computer**: A small, single-user computer based on a microprocessor.
- **Workstation**: A powerful, single-user computer. A workstation is like a personal computer, but it has a more powerful microprocessor and, in general, a higher-quality monitor.
- **Minicomputer**: A multi-user computer capable of supporting up to hundreds of users simultaneously.
- **Mainframe**: A powerful multi-user computer capable of supporting many hundreds or thousands of users simultaneously.
- **Supercomputer**: An extremely fast computer that can perform hundreds of millions of instructions per second.

a) SUPERCOMPUTER

Supercomputer is a broad term for one of the fastest computers currently available. Supercomputers are very expensive and are employed for specialized applications that require immense amounts of mathematical calculations (number crunching). For example, weather forecasting requires a supercomputer. Other uses of supercomputers are scientific simulations, (animated) graphics, fluid dynamic calculations, nuclear energy research, electronic design, and analysis of geological data (e.g. in petrochemical prospecting). Perhaps the best known supercomputer manufacturer is Cray Research.

b) MAINFRAIME COMPUTER

Mainframe is a very large and expensive computer capable of supporting hundreds, or even thousands, of users simultaneously. The chief difference between a supercomputer and a mainframe is that a supercomputer channels all its power into executing a few programs as fast as possible, whereas a mainframe uses its power to execute many programs concurrently. In some ways, mainframes are more powerful than supercomputers because they support more simultaneous programs. But supercomputers can execute a single program faster than a mainframe. The distinction between small mainframes and minicomputers is vague, depending really on how the manufacturer wants to market its machines.

c) MINICOMPUTER

It is a midsize computer. In the past decade, the distinction between large minicomputers and small mainframes has blurred, however, as has the distinction between small minicomputers and workstations. But in general, a minicomputer is a multiprocessing system capable of supporting from up to 200 users simultaneously.

d) WORKSTATION

It is a type of computer used for engineering applications (CAD/CAM), desktop publishing, software development, and other types of applications that require a moderate amount of computing power and relatively high quality graphics capabilities. Workstations generally come with a large, high-resolution graphics screen, at large amount of RAM, built-in network support, and a graphical user interface. Most workstations also have a mass storage device such as a disk drive, but a special type of workstation, called a diskless workstation, comes without a disk drive. The most common operating systems for workstations are UNIX and Windows NT. Like personal computers, most workstations are single-user computers. However, workstations are typically linked together to form a local-area network, although they can also be used as stand-alone systems.

N.B.: *In networking, workstation refers to any computer connected to a local-area network. It could be a workstation or a personal computer.*

e) **PERSONAL COMPUTER**

It can be defined as a small, relatively inexpensive computer designed for an individual user. In price, personal computers range anywhere from a few hundred pounds to over five thousand pounds. All are based on the microprocessor technology that enables manufacturers to put an entire CPU on one chip. Businesses use personal computers for word processing, accounting, desktop publishing, and for running spreadsheet and database management applications. At home, the most popular use for personal computers is for playing games and recently for surfing the Internet.

1.2.2 History of Computers

Computers of specific electronic era are often referred to as a generation. Each generation is characterized by a major technological development that fundamentally changed the way computers operate, resulting in increasingly smaller, cheaper, more powerful and more efficient and reliable devices.

Definition: *A computer generation is an era or period in the history of computers in which the computer was characterized by a major technological development that fundamentally changed the way computers operate*.

a) **First Generation (1942-1955)**
First generation computers were made using vacuum tubes. A vacuum tube was a fragile glass device that could control and amplify electronic signals. These computers were very large taking up entire rooms, very expensive to operate and in addition to using a great deal of electricity, generated a lot of heat, which was often the cause of malfunctions.
First generation computers relied on machine language, the lowest-level programming language understood by computers, to perform operations, and they could only solve one problem at a time. Input was based on punched cards and paper tape, and output was displayed on printouts. Some examples of first generation computers are ENIAC (**E**lectronic **N**umerical **I**ntegrator **A**nd**C**alculator), EDSAC (**El**ectronic **D**elay **S**torage **A**utomatic **C**alculator) and UNIVAC I (**U**niversal **A**utomatic **C**omputer I).

b) **Second Generation (1955-1964)**
Transistors replaced vacuum tubes and ushered in the second generation of computers. The transistor was far superior to the vacuum tube, allowing computers to become smaller, faster, cheaper, more energy-efficient and more reliable than their first-generation predecessors. Though the transistor still generated a great deal of heat that subjected the computer to damage, it was a vast improvement over the vacuum tube. Second-generation computers still relied on punched cards for input and printouts for output.
Second generation computers moved from cryptic binary machine language to symbolic, or assembly languages, which allowed programmers to specify instructions in words. High-level programming languages were also being developed at this time, such as early versions of COBOL and FORTRAN. These were also the first computers that stored their instructions in their memory, which moved from a magnetic drum to magnetic core technology.

c) **Third Generation (1964-1975)**
Third generation computers were designed with the use of integrated circuits (ICs). Integrated circuits made it possible to embed a large number of transistors into very small surface area of silicon known as chip. Instead of punched cards and printouts, users interacted with third generation computers through keyboards and monitors and interfaced with an operating system, which allowed the device to run many different applications at one time with a central program that monitored the memory. Computers for the first time became accessible to a mass audience because they were smaller and cheaper than their predecessors.

d) **Fourth Generation (1975)**

The microprocessor brought the fourth generation of computers, as thousands of transistors could be placed on a single silicon chip. The term very large scale integration (VLSI) was coined to describe this technology. What in the first generation filled an entire room could now fit in the palm of the hand. As these small computers became more powerful, they could be linked together to form networks, which eventually led to the development of the Internet. Fourth generation computers also saw the development of GUIs, the mouse and handheld devices.

e) **Fifth Generation (present)**

Till fourth generation computers, the main stress was on hardware technology. Fourth generation computers are faster, more accurate, reliable, smaller in size and very cheap, still they lack thinking power. Fifth generation computing devices are based on artificial intelligence. Artificial intelligence will give computers thinking power and capability to make decisions like human beings. They are still in development though there are some applications, such as voice recognition, that are being used today.

1.2.3 APPLICATIONS OF COMPUTERS

a) Business: Computer is used in business organizations for: Payroll calculations, Budgeting, Sales analysis, Financial forecasting, Managing employees database, Maintenance of stocks etc.

b) **Banking**: Banks provide following facilities: Banks provide online accounting facility, which includes current balances, deposits, overdrafts, interest charges, shares, and trustee records. ATM machines are making it even easier for customers to deal with banks.

c) **Insurance**: Insurance companies are maintaining a database of all clients with information showing: procedure to continue with policies, starting date of the policies, next due installment of a policy, maturity date, interests due, survival benefits, and bonus.

d) **Education:** The computer has provided a lot of facilities in the education system. The computer provides a tool in the education system known as CBE (Computer Based Education). CBE involves control, delivery, and evaluation of learning. The computer education is rapidly increasing the graph of number of computer students. There are number of methods in which educational institutions can use computer to educate the students. It is used to prepare a database about performance of a student and analysis is carried out on this basis.

e) **Marketing**: In marketing, uses of computer are following:

- **Advertising -** With computers, advertising professionals create art and graphics, write and revise copy, and print and disseminate ads with the goal of selling more products.
- **At Home Shopping -** Home shopping has been made possible through use of computerized catalogues that provide access to product information and permit direct entry of orders to be filled by the customers.

1.2.4 COMMUNICATIONS AND INFORMATION SYSTEMS

1.2.4.1 COMMUNICATION SYSTEMS

When participants within the information system have a *need to transmit and receive data or information*, the type of system required is a communication system. Communication systems support people who are working together, by *enabling the exchange of data and information electronically*.

a) **Components of a Data communication system**

A Communication system has following components:

1. **Message**: It is the information or data to be communicated. It can consist of text, numbers, pictures, sound or video or any combination of these.

2. **Sender**: It is the device/computer that generates and sends that message.

3. **Receiver**: It is the device or computer that receives the message. The location of the receiver computer is generally different from the sender computer. The distance between sender and receiver depends upon the types of network used in between.

4. **Medium**: It is the channel or physical path through which the message is carried from sender to the receiver. The medium can be wired like twisted pair wire, coaxial cable, fiber-optic cable or wireless like laser, radio waves, and microwaves.

5. **Protocol**: It is a set of rules that govern the communication between the devices. Both sender and receiver follow same protocols to communicate with each other.

b) Examples of Communication systems:

(i) Teleconferencing: The use of an electronic transmission to allow a meeting to occur at the same time in different locations. Teleconferencing as an alternative to a face to face meeting; how audio data is transmitted and received, how video data is transmitted and received provides us with advantages such as the reduction in costs and disadvantages such as the removal of inter-personal relationships. Some application areas of teleconferencing are business use, distance education.

(ii) Video-Conference: A meeting that allows people in different locations to see video images of each other on screen, as well as hear speech.

(iii) E-mail (Electronic Mail): Allows communication with other email users by sending and recording electronic messages using a computer. Email was one of the earliest uses of the Internet and is now widely used. It is a fast, economical and convenient way to send messages to people all over the world. Characteristics of email includes: sender details, receiver details, a subject, a message, date.

(iv) Telephone: A system for transmitting sounds or speech between distant locations along telephone lines.

1.2.4.2 INFORMATION SYSTEMS

Information systems are computer systems that support end users, giving them access to the information. For a large number of information systems, the data is held in databases and access is via database management systems.

(a) Computerised School Based Management System: A computerised school based management system consists of a computer holding and processing information about the school. This system is expected carry out the following:

- Record the attendance details of students on daily basis at the opening and closing, this will generate the names of students with poor attendance so that such students can be punished.
- Keep students' personal and staff information such as names, date of birth, sex, class, guardians, medical records etc…
- Generates various lists such as class list, school fees payment list etc…
- Generate information and statistics to manage examinations of the school.
- Help prepare timetable, meeting reports, teaching hours of the school.

(b) Employee Record and Payroll Processing System: Computers are used in managing employee records and payroll since companies employ many workers. Payroll processing is made easy as automatic calculations can be made for several employees at a given time.

(c) Library Management Information System: A typical library contains thousands of books, magazines, reports and other documents which users may wish to borrow. Maintaining records of books and borrowers are well suited with computer systems. Computer systems make it easier for library personnel to answer queries about status (Title, Author, ISBN Number) of books more easily than a manual system.

(d) Stock Control Systems: A stock control system enables a user to manage his stock more effectively. A file is created for each item and for each item is assigned a unique identifier (usually a number), this identifier is used in all transactions concerning the item.

(e) Reservation Systems: There are different types of reservation systems:
(i) **Flight Reservation**: Airlines use computers to keep records of flights, seat availability, passengers, crew members and other employees. By entering the flight number and date, a reservation clerk can immediately inform a passenger whether or not seats are available. If a seat is not available, the computer can quickly give the closest date on which seats are available. All flights and passenger records are stored on a computer system in one location. Travel agencies all over the world can access this data via communication links (telephone or internet). This makes it very easy to confirm (or deny) a seat reservation.
(**ii**) *Hotels, theatres* and *car rental agencies* all use computer systems to keep records of room availability, seat availability and car availability respectively.

1.2.5 Commercial and General Data Processing Systems

a) **ATM Systems**

ATM stands for Automatic/Automated Teller Machine. It is a machine that is connected to a bank's computer system that provides the bank's customers with access to financial transactions in a public space without the need for a cashier, human clerk or bank teller. The ATM can be used by the customers to make cash withdrawals, credit card advances or check their account balances.

On most modern ATM systems, a customer is identified by inserting a plastic ATM card with a magnetic stripe or a plastic smart card with a chip, which contains a unique card number and some security information. Authentication is provided by the customer entering a *personal identification number* (PIN). Upon successful entry of the PIN, the customer may perform a transaction.

ATMs are known by various other names including *automatic/automated banking machine* (ABM), *automated transaction machine* and cashpoint.

b) Stock Control Systems:

The collection of items that a business manufactures or sells is called stock. In a shop for example, the stock includes all of the items on the shelves and out the back in the storeroom.

It is important that a business does not keep too much stock or too little stock. This is because:

- ✓ Too much stock costs money as you have to store it all somewhere
- ✓ Too much perishable stock (e.g. food) means that it may go bad before it is sold
- ✓ Too little stock means that you might run out of stock before the next delivery arrives

Definition: *A stock control system is basically a database that keeps track of stock and informs users of when to re-order along with helpful sales reports.*

When items are sold or delivered, their codes are input to the system either manually or using a point-of-sale terminal (barcode scanner or similar technology). The database matches the codes with the items' names and prices, prints an itemized bill and uses the data to update stock levels. It also updates a sales file which can be used there and then to calculate all sorts of statistics.

Stock control systems make it very easy for stock levels to be monitored, and for stock to be reordered when it is running low.

c) EFTPOS Systems:

EFTPOS stands for *Electronic Fund Transfer at the Point Of Sale*. It is a system that combines a business' stock control system (EPOS) with an electronic fund transfer system, thereby enabling the business to conduct financial transactions electronically. It allows individuals to pay for goods using credit or debit cards.

To pay for goods, a customer inserts their debit or credit card into a terminal device and type in a PIN number to verify that it is their card. The system electronically contacts their bank to check that the card is valid and hasn't been stolen and also that there is enough fund in their account to pay for the goods. Once payment has been authorized by the bank, money is removed from the customer's account and electronically paid into the shop's account.

1.2.6 Industrial, Scientific and Technical Uses of computer systems

a) Simulation and Modelling

Modelling and simulation (M&S) is getting information about how something will behave without actually testing it in real life. For instance, if we wanted to design a race car, but weren't sure what type of spoiler (a device on a car that is positioned so that it stops the air from flowing around the vehicle in a smooth and so helps to control it) would improve traction the most, we would be able to use a computer simulation of the car to estimate the effect of different spoiler shapes on the coefficient of friction in a turn. We're getting useful insights about different decisions we could make for the car without actually building the car.

Why use computer models?

As well as financial modelling, spreadsheet software can be used for many other kinds of computer model:

- modelling supermarket queues
- modelling the stresses which will be borne by a new bridge
- modelling traffic flow in a new road system

Computer models are cheaper to setup than alternative methods that could be used to predict what will happen in a system, eg building a prototype. Other benefits include being able to:

- make alterations and quickly see the outcomes
- repeat tests several times over
- learn from "what if?" scenarios
- model dangerous situations safely

b) Computer-Aided Design

Computer-aided design (CAD) is the use of computer technology in the design process. A CAD software package allows a designer to create technical drawings and schematics which can be 2-dimensional or 3-dimensional.
The benefit of CAD software packages is their ability to provide a digital prototype of the product at early stages of the design process, which can be used for testing and evaluation. Examples of CAD applications are AutoCAD used for technical drawings and ARES used for designing printed circuit boards.

c) Computer-Aided Manufacturing

Computer-aided manufacturing (CAM) is the use of computer technology to assist in the manufacturing process. In CAM, the computer is used to program, direct, and control production equipment in order to manufacture products. Its primary purpose is to create a faster production process, components and tooling with more detailed dimensions and material consistency, which in some cases, uses only the required amount of raw material (thus minimizing waste), while simultaneously reducing energy consumption. For example, on the

production line of a car plant, computers will control the robots that spot-weld the car body together or the robots that spray-paint the car.

d) Global Positioning System (GPS). An accurate worldwide navigational and surveying facility based on the reception of signals from an array of orbiting satellites.The GPS indicates an object's position in space. The GPS is a satellite-based system capable of determining location.

1.2.7 Control Systems, Embedded Systems and Robotics

a) **Control and Monitoring Systems**

Control is the process of monitoring activities to ensure that they are being accomplished as planned and of correcting any significant deviations. A computer control system is a computerized system designed to control a process. Examples are traffic light control systems, greenhouse control systems, and patient monitoring systems.

1. **Green House Control System**

In a greenhouse system, a computer controls the temperature in a greenhouse to maintain the conditions required for the plants in it to grow. The greenhouse has temperature and humidity sensors linked to a computer, and the computer has a control program storing details of the correct temperature and humidity settings. The greenhouse is fitted with a heater, sprinkler and window motor, also linked to the computer. If the humidity falls below the values stored in the program, the computer activates the sprinklers and closes the windows. If the temperature falls outside the values stored in the program, the heater is activated by the computer.

The system monitors the conditions night and day with immediate response to any changes. To alter the growing conditions the values in the computer program can of course be changed.

2**. Traffic Control System**

In a traffic light system a computer controls the sequences of lights displayed at a cross-roads to ensure that cars do not crash. Additionally the computer operates a pedestrian crossing to let pedestrians cross the road when a button is pressed.

3. **Patient Monitoring System**

In a patient monitoring system, a computer controls the monitoring system used to measure the health condition of patients. If a patient's condition gets critical, the system alerts the nurses or a doctor.

b) Embedded Systems

An embedded system is a small computing device (microcontroller) that is built into a larger equipment often as a single chip and dedicated to a given task. They control many devices in use today such as digital watches, mobile phones, microwave ovens, washing machines, vehicles, photocopiers, and very large stationary installations like traffic lights, factory controllers or the systems controlling nuclear power plants.

c) Robotics

Robotics is the branch of technology that deals with the design, construction, operation and application of robots. A robot is a machine that is designed to repeatedly do what humans can do with speed and precision. An important application of this technology has been to create robots to perform certain functions that are dangerous for human beings, or to do tasks that can be more effectively performed by machines than by people. Although it may not have the physical appearance of a human being, a robot may be thought of as a machine acting as a person while being controlled by a computer.

Drill Questions:

1. What does an ICT system consist of?
(a) Hardware, data, networks, people (b) Hardware, software, data, people
(c) Hardware, people, storage, software

2. At what speed do ICT systems transmit and receive information?
(a) Really slowly (b) Rapidly
(c) Quite quickly

3. Which of these is an example of a communications technology?
(a) Printer (b) Scanner (c) The internet

4. How do ICT systems differ from manual systems?

(a) Less productive (b) About the same (c) More productive

5. What is an information system's main management task?
(a) Data and information (b) Data and people (c) Data and machinery

6. Which of these is an output from a control system?

(a) Paper reports on industrial processes (b) Formulae from a spreadsheet
(c) Instructions sent to machinery or equipment

7. What is the process known by in which output affects the input?

(a) Processing (b) Feedback (c) Media integration

8. What manages successful data transmission from one part of the system to another?
(a) Processing system (b) Communication system (c) Information system

9. In what order do ICT systems deal with data?
(a) Process, input, output (b) Input, output, process (c) Input, process, output

10. What is a short acronym that highlights the importance of accurate data in an ICT system?

(a) GAGO (b) GIGI (c) GIGO

11. A company decides to introduce robots to the production line to manufacture cars. Discuss four advantages and four disadvantages to the workers of introducing robots rather than using humans to manufacture cars.

12. What is Telemedicine? Name two benefits of telemedicine to remote communities in Cameroon. (**Q9(iv), CGCE2015**)

13. Multinational companies often use video-conferencing. Discuss four advantages and disadvantages of holding video-conferencing rather than face-to-face meetings.

14. Discuss three possible effects on people who live in some developing countries of having no access or very limited access to ICT. (**Q3(iii), CGCE2015**)

15. Explain the term *Embedded system.* Give two examples of devices which might have an embedded system.

16. A microprocessor embedded in a plastic can be used to read or store information on the card using special terminals.

(a) Outline one advantage and one disadvantage of such cards.

(b) Outline two conditions that needs to be met for such cards to be widely used.

17. Briefly explain the following as used in ICT, giving an example each; monitoring system, Control systems, automated systems, embedded systems. (**Q8(i), CGCE2015**)

18. Explain in brief the various generations in Computer Technology.

19. How useful is a computer to you?

20. Give TWO uses of each of the following information systems. Your answers should include the type of data or information they manage, and also state how useful the information is to the client organisation. (a) Patient record system (b) Payroll system (**Q7(i),CGCE 2014**)

21. Explain, using examples, the following computer crimes. (a) Simulation. (b) Videoconferencing (**Q1(i),CGCE 2014**)

22. (a) Describe simulation and a situation that can be situation that can be simulated.

(b) Give three advantages of simulation to architects (**Q6(ii),CGCE 2016**)

23. Explain the following computer terms giving examples where necessary:
(a) Videoconferencing (b) Stock Control system (**Q1(i), CGCE2016**)

24. Discuss with examples three changes in the workplace due to the increased use of ICT. Your answer should refer to the changes for the employees. (**Q3(iii), CGCE2016**)

CHAPTER 2: THE SOCIAL, LEGAL, ETHICAL AND ECONOMIC IMPLICATIONS OF THE USE OF COMPUTERS

2.1 Introduction

Computers have had many effects on individuals in society. They have impacted on the way individuals work, socialize and run their lives. As a result of computers, an individual's values are now in constant flux. The moral and ethical framework that guides an individual is constantly changing as is the economic and legal framework within which lives are led.

2.2 Social and Economic Effects of Computers

i) ***Reduction of Manufacturing Jobs***: the increasing sophistication of computers has allowed machines with robotic arms to perform complex tasks with the dexterity that once required humans. These machines, over the long run, were cheaper for companies to use than the human workers for whom the employers had provided benefits and retirement plans. This has reduced a traditionally reliable source of employment for workers.

ii) ***Piracy of Copyrighted Materials***: Before computers, piracy of movies and music was a small problem. The greatest threat the music industry faced was the ability of stereo systems to record songs playing on the radio onto cassette tapes. However, computers and the Internet provided a new venue through which a single digitized copy of a movie or song could be infinitely distributed to other computers across the world. This caused a drop in music and video sales, and created a small black market of cheap bootleg copies of these pirated items.

iii) ***Financial Engineering***: Computers, and the computational power they represent, enabled the rise of the financial entities known as hedge funds. These are companies of brokers who design sophisticated computer models to analyse market data in real time to predict movement trends. The computers use these calculations to conduct instant transactions electronically, generating higher market returns than brokers or portfolio managers ever saw before the computers' deployment.

iv) ***Commerce or tertiary service industry***, has been changed a great deal by the use of computers.

v) ***Databases***: data was once kept in paper or card filing systems where it was very difficult to summarise and manipulate. Once captured by a computer system data can be made available to anyone (legally or illegally) and it can be processed to reveal information, patterns and trends that would remain hidden in paper-based systems.

vi) ***Financial trading***: money can be made by dealing in stocks and currencies. A tiny and brief change can be exploited by a trading system that is fast enough to react to such changes and money can be made on the difference. The faster a computer system the better the opportunities to make money in such trading so financial organisations compete to have the fastest and most powerful systems.

vii) The automobile industry has lost labour as people have been replaced by robots. Much of this would have been classed as semi-skilled or skilled manual labour. Without further education and training the people displaced will find it hard to find comparable work.

vi) There is reduction in working hours as a result of automation of some tasks which can lead to reduction of salaries.

vii) The introduction of computers has led to retraining or reskilling as many employees have to be trained on how to use a computer.

ix) Automation of tasks can lead to difficult jobs becoming easy so anyone can do them. As a result, a skilled employee suddenly is the proud possessor of skills that no-one needs any more.

x) Automation of tasks has led to an increase in productivity thereby improving the profit of the organization.

The post office has become more of a bill paying center than a communication center as it is now faster and cheaper to communicate online through emails and chats.

2.3 SYSTEM SECURITY AND RELIABILITY

a) System Security: An (operating) system is responsible for controlling access to system resources, which will include sensitive data. The system must therefore include a certain amount of protection for such data, and must in turn control access to those parts of the system that administer this protection. System security is concerned with all aspects of these arrangements. It is literally a means or method by which something is secured through a system of interworking components and devices. Examples of how system security works:

i) **Home security systems** work on the simple concept of securing entry points into a home with sensors that communicate with a control panel or command centre installed in a convenient location somewhere in the home.

ii) **Control Panel:** The control panel is the computer that arms and disarms the security systems, communicates with each installed component, sounds the alarm when a security zone is breached, and communicates with an alarm monitoring company.

iii) **Door and Window Sensors:** Door and window sensors are comprised of two parts installed adjacent to each other. One part of the device is installed on the door or window and the other on the door frame or window sill. When a door or window is closed, the two parts of the sensor are joined together, creating a security circuit.

b) System Reliability: Reliability is an attribute to a computer related component (hardware, or software, or network for example) that consistently performs according to its specifications. It has long been considered one of three related attributes that must be considered when making, buying or using a computer product or component. Thus reliability requires features that help and avoid faults; not just to run quietly with uncorrupted data but to detect and correct faults where possible e.g repeat an operation or isolate the fault and report it.

c) System Resilience: resilience is the long term capacity of a system to deal with change and continue to develop. The term resilience is used differently by different communities. In general engineering systems, fast recovery from a degraded system state is often termed as resilience.

d) Privacy and Integrity of Data

To ensure data is safe from hackers (e.g by passwords, firewalls) and data is protected from corruption.

i) Data Integrity is about ensuring that data are correct. Data might be wrong on input; or it may be corrupted accidentally by hardware failure or deliberately by a hacker. Data must be accurate and reliable or else it isn't any use. Validation and verification are used to ensure that data are correct when entered.

ii) Data Privacy: as the amount the amount of stored data has increased it has become increasingly important to have the means and the legal framework to keep it private. Government data: military, intelligence, diplomatic, economic… Personal: address, phone number, salary…Commercial: products, services, prices, processes, performance, plans…

iii) Data Security: it is concerned with keeping data safe from events such as theft. Damage, flood or fire.

e) Safe working practices:

i) Use passwords that can't be easily guessed and protect your passwords

ii) Minimize storage of sensitive information

iii) Beware of scams

iv) Protect information when using the internet and email

v) Make sure your computer is secured with an antivirus and all necessary security "patches" and updates.

vi) Secure laptop computers and mobile devices at all times; Lock them up and carry them with you.

vii) Shut down, log off, lock or put your computer or other devices to sleep before leaving them unattended and make sure they require a secure password to start up or wakeup

viii) Don't install or download unknown or unsolicited programs/apps.

ix) Secure your area before leaving it unattended

x) Make backup copies of files or data you are not willing to lose.

2.4 Ergonomics Design

a) Definition: The applied science of equipment design, as for the work place, intended to maximize productivity by reducing operator fatigue and discomfort. In other words it is the study of people's efficiency in their working environment.

b) Computer Setup:

1. Use a good chair with a dynamic chair back and sit back.
2. The eye-level should be the same as the level of the monitor. You should be able to see the contents in the monitor without bending your neck.
3. No glare on screen, use an optical glass anti-glare filter where needed.
4. Sit at arm's length from monitor as a good viewing distance.
5. Feet on floor or stable footrest.
6. Use a document holder, preferably in-line with the computer screen.
7. Wrists flat and straight in relation to forearms to use keyboard/mouse/input device.
8. Arms and elbows relaxed close to body.
9. Top of monitor casing 2-3" (5-8 cm) above eye level.
10. Use a negative tilt keyboard tray with an upper mouse platform or downward tiltable platform adjacent to keyboard.
11. Center monitor and keyboard in front of you.
12. Use a stable work surface and stable (no bounce) keyboard tray.

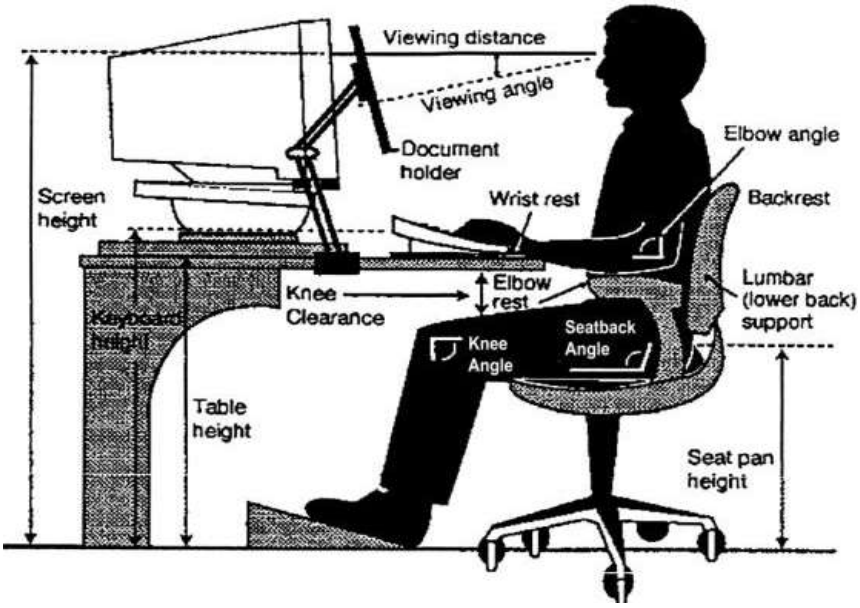


Figure 2.1: *Ergonomic setup*.

i) Using Keyboard

Keep the keyboard flat. The hands and the keyboard should be parallel and perpendicular. Do not use the built-in tips that elevates the back of the keyboard.

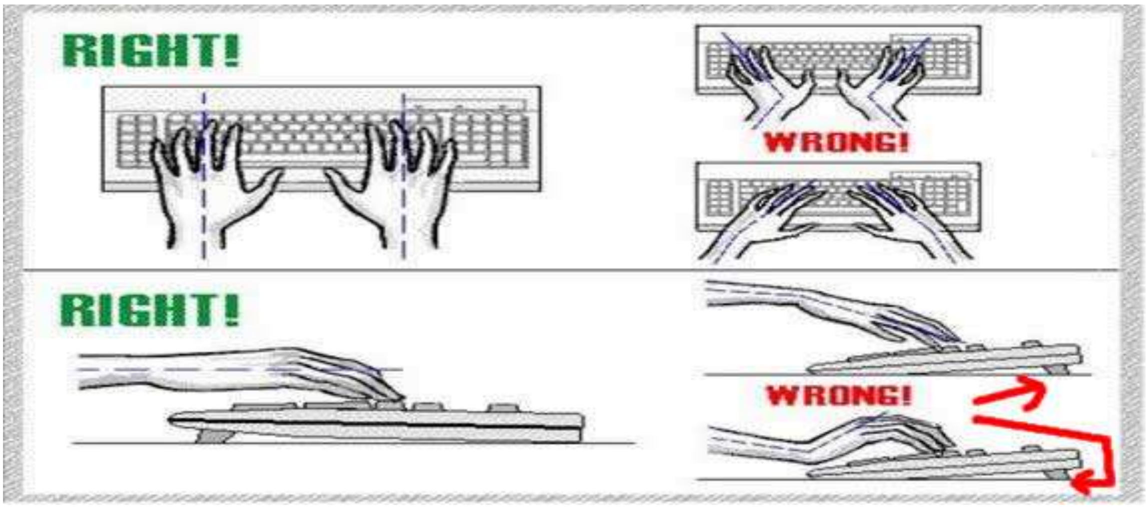


Figure 2.2: *image showing the right position of fingers on a keyboard*

ii) Using Mouse

Switch hands for using the mouse periodically. Keep the hand and wrist straight.

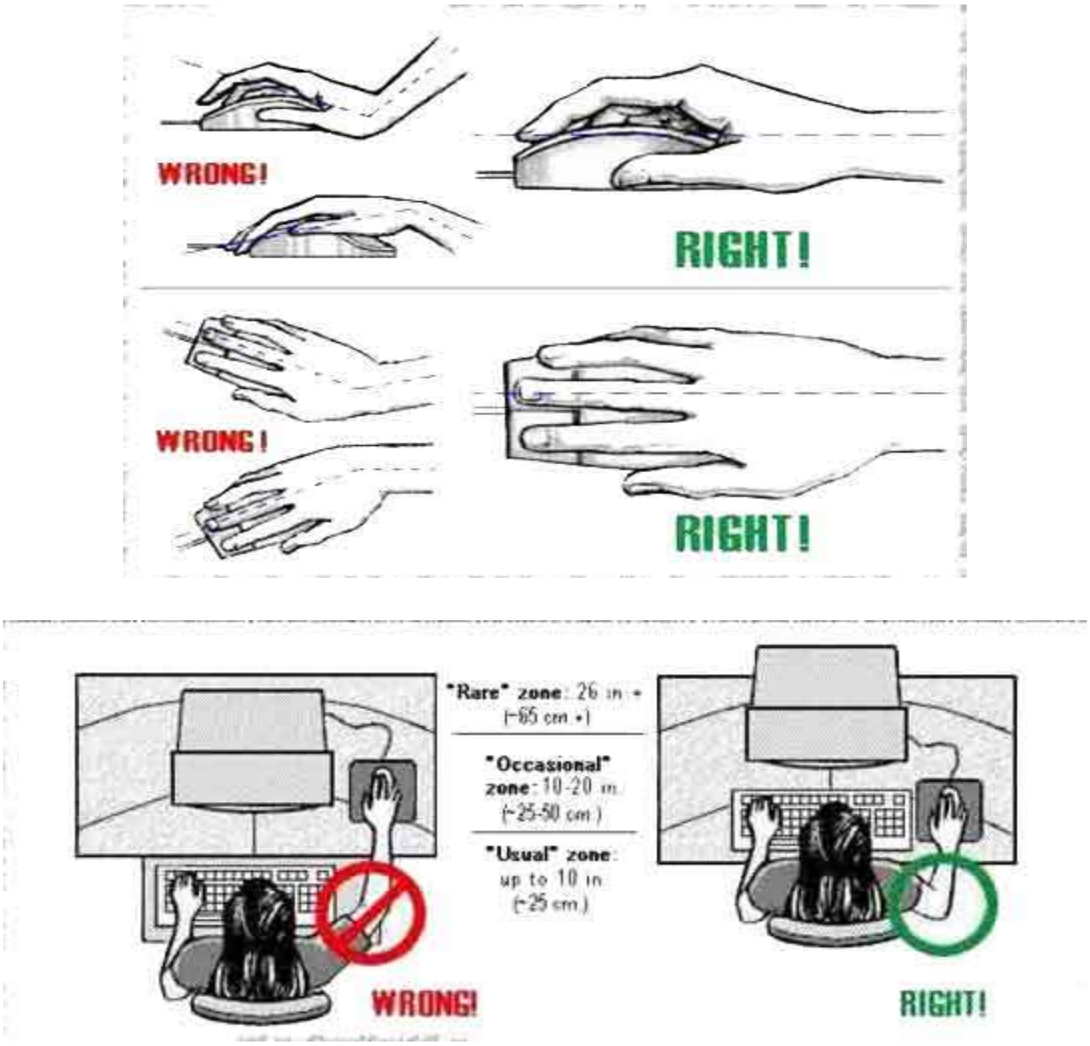


Figure 2.3: *images showing the right way to use a mouse*

iii) Stretching:

- Stretch inside of your body - drink water often.
- Take a walk every once in a while - e.g. every 45 minutes or so.

c) Some Computer related diseases, symptoms and their possible remedies:

i) Repetitive Strain Injury (RSI)

Body movements are produced by contracting and relaxing muscles. The muscles are attached to bones by tendons. Tendons are smooth, and in some parts of the body they glide back and forth inside tubes called synovial sheaths. RSI results when repeated stress is placed on the tendons, muscles, or nerves of the body, causing inflammation or damage.

***Definition**: RSI is a health problem resulting from overusing a part of the body to perform a repetitive task, like typing and clicking, thereby causing trauma to that part.*

RSI is also called, ***cumulative trauma disorder** (**CTD**)*, repetitive strain disorder, repetitive stress injury, repetitive stress disorder, overuse syndrome, and musculoskeletal disorder.

Some of the most common types of RSI are:

a) *Tendonitis* - inflammation of the tendons
b) *Tenosynovitis* - inflammation of the synovial sheath
c) *Carpal Tunnel Syndrome* - results when the median nerve is compressed, either from the swelling of tendons and sheaths or from repeated bending of the wrist

Some of the conditions that may lead to CTDs are:

i. *Repetition* - long or concentrated hours of typing or using a mouse
ii. *Posture* - long hours of sitting in the same position while typing, especially if it is in an uncomfortable or poorly supported position, or if the wrists are bent
iii. *Lack of rest* - intensive hours at the keyboard with few breaks

Symptoms of CTDs:

- Tingling or numbness in the hands or fingers
- Pain in fingers, hands, wrists, or even shooting up into the arms or forearms
- Loss of strength or coordination in the hands
- Numbness or discomfort in the hands which wakes you up at night

ii) Carpal Tunnel Syndrome

Carpal tunnel syndrome (CTS) is a compression of the median nerve in the wrist. Symptoms include pain and numbness in the hand (especially at night), clumsiness, paresthesia (pins and needles), and trophic changes (such as muscle wasting). In a true CTS, these are felt where the median nerve goes: the palm side of the index and middle fingers and part of the thumb and

ring finger. Conservative treatment without surgical intervention will usually give relief, especially if done early after onset.

Similar symptoms can also be due to nerve compression in the neck, shoulder or arm from such things as tight neck or shoulder muscles (ie: thoracic outlet syndrome or pectoral muscle contracture) or poor neck mechanics to name a few. These other problems are often mis-diagnosed as CTS. To help see if you have a true carpal tunnel syndrome or not, use Phalen's test.

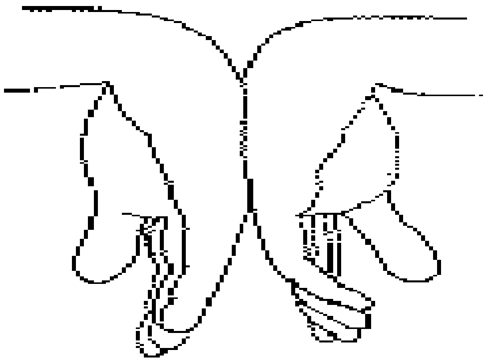


Figure 2.4: *Phalen's Test*

Place the backs of both of your hands together and hold the wrists in forced flexion for a full minute. (Stop at once if sharp pain occurs). If this produces numbness or "pins and needles" along the thumb side half of the hand, you most likely have Median nerve entrapment (Carpal Tunnel Syndrome). Examination by a health care professional familiar with these conditions is the way to be sure of the diagnosis and get proper treatment.

- Keyboards: Be sure to get the height right to prevent too much bend at the wrist and allow the forearm to have some support. The arms should hang loose to prevent the shoulder muscles from cramping. Many keyboards can tilt; unfortunately, most of them tilt the wrong way. If anything the keyboard should tilt to help the wrist stay straight, which is to say raising the space bar end and lowering the "top"(the F1, F2 etc.) end. Tilting the key board the other way, (space bar lower and "top" row higher) can set you up for carpel tunnel syndrome.

Treatment Effective conservative treatment of CTS should include:

- Chiropractic manipulation of the wrist, forearm and hand
- Ice massage (10 to 12 minutes) several times a day
- minimizing any irritating activities
- wrist strengthening exercises
- wrist stretching exercises
- possible use of wrist brace or splint while sleeping
- applying sound ergonomic principles (see The Human Interface)

Seeing your Chiropractor for a check up to keep the joints mobile and mechanically well aligned is a great aid. These treatments can minimize the formation of CTS as well as

decreasing its pain and impact on your job and lifestyle. While seeing your Chiropractor, ask for more details about what is best for you, personally, regarding work habits, exercise and stretching to promote good health and in the prevention and /or management of carpal tunnel syndrome.

iii) Computer Electromagnetic Radiation

Your health may be effected by radiation given off by your computer. Computers do generate very low levels of infrared light, visible light, ultraviolet light, X-rays and electromagnetic fields. Recent studies have raised some concerns about ELF and VLF waves, PCBs, ultrasound, electrostatic fields and other emissions as well.

Extremely low frequency (ELF) electromagnetic waves are produced by high voltage power lines, transformers, virtually everywhere electricity flows, including a VDT screen. Very low frequency (VLF) electromagnetic waves are also emitted by computers.

Much of the current, data regarding how much radiation is safe, is inconclusive or even contradictory. After much searching and due consideration, I suggest the following prudent course of action.

Recommendations:

1. *Keep your distance.* Electromagnetic radiations follows the inverse square rule, which is to say the further away you are from the source, the weaker they get and they do so quickly. You can protect yourself with space. I recommend you stay at least 75 centimeters (30 inches) from your terminal and at least one meter (40 inches) from other terminals.

2. *Keep it fixed*. X-ray production increases dramatically when the VDT is damaged, improperly maintained, or just plain worn out. PCBs are sometimes released by very old VDT models (ie built before 1970).

3. *Limit your time.* If you have to sit at the computer for hours every day, you should seriously consider limiting how much time you spend playing video games and watching TV at home. This is most especially true if you are pregnant.

iv) **Computer Vision Syndrome**

This comprises of problems related to seeing correctly like visual fatigue, dry itchy and sore eyes, blurred or double vision, burning watery eyes and loss of colour in affected regions. To avoid eyestrain, take the following precautions:

- Exercise your eyes periodically focusing on objects at different distances
- Blink regularly
- Position the monitor to avoid glare
- Keep your monitor clean
- Service, repair or replace a monitor that flickers

2.5 Computer Crimes

a) **Definition:** Alternatively referred to as *cybercrime*, *e – crime, electronic crime* or *Hi-tech crime*. Computer crime is an act performed by a knowledgeable computer user, sometimes referred to as a *hacker* that illegally browses or steals a company's or individual's private information. In some cases, this group of individuals may be malicious and destroy or otherwise corrupt the computer or data files.

b) **Types of computer Crimes:**

i) **Child Pornography**: This is material showing children in erotic poses or having sex. It is usually graphic material in the form of drawings, photographs or video, but can be in writing as well. Child pornography that involves real children is a record of child sexual abuse. Usually, these children are shown or described as being in different stages of undress, with some clothes off, or completely naked.

ii) **Cyber terrorism**: Cyber terrorism can be defined as an act of terrorism (Hacking, threats and blackmailing towards a business or person) committed through the use of cyberspace or computer resources. As such, a simple propaganda in the Internet, that there will be bomb attacks during the holidays can be considered cyber terrorism

iii) **Cyberbullying or Cyber stalking**: Cyber stalking is a crime in which the attacker harasses a victim using electronic communication, such as e-mail or instant messaging (IM), or messages posted to a web site or a discussion group. Cyber stalking messages differ from ordinary spam in that a cyber-stalker targets a specific victim with often threatening messages, while the spammer targets a multitude of recipients with simply annoying messages.

iv) **Creating Malware**: Writing, Creating or Distributing Malware (e.g Viruses and spyware.)

v)**Denial of Service Attack**: A denial or degradation of service (DoS) is an attack to a computer system that puts it out of action by overloading it with data in a way that the system was never prepared to handle. A DoS attack makes the system unavailable to its intended users. A **distributed denial-of-service** (DDoS) attack is one in which a multitude of compromised systems attack a single target, thereby causing denial of service for users of the targeted system.

vi) **Espionage**: Spying on a person or business.

vii) **Fraud**: Manipulating data eg changing banking records to transfer money to an account. Computer Fraud is also defined as any act using computers, the internet, internet devices, internet services, to defraud people, companies, or government agencies of money, revenue, or internet access. There are many methods to perform these illegal activities. Phishing, social engineering, viruses and DDos attacks are fairly well known tactics used to disrupt service or gain access to another's funds, but this list is not inclusive.

viii) **Harvesting**: Collect amount or other account related information on people.

ix) **Identity Theft**; pretending to be someone you are not.

x) **Intellectual property theft**: stealing another person or company intellectual property

xi) **Phishing**: Deceiving individuals to get private or personal information about that person.

xii) **Salami Slicing**: Stealing tiny amounts of money from each transaction

xiii) **Spamming**: Distributed unsolicited e-mail to dozens or hundreds of different addresses

xiv) **Spoofing**: Deceiving a system into thinking you are someone you really are not.

xv) **Unauthorized access**: Gaining access to systems you have no permission to access. This is when someone gains access to a website, program, server, service, or other system using someone else's account or other methods. For example, if someone kept guessing a password or username for an account that was not theirs until they gained access it is considered to be unauthorized.

xvi) **Wiretapping**: connecting a device to a phone to listen to conversations

c) **Malware attacks**

Malware (malicious software) is any software that could harm a computer system, interfere with a user's data, or make the computer to perform actions without the owner's knowledge or permission. Examples are virus, worms, Trojan horse, spyware and logic bombs.

i) Virus: A virus is a computer program that can copy itself and infect a computer where it destroys files and disrupts the operation of the computer. A virus can spread from one computer to another (in some form of executable code) when its host is taken to the target computer.

ii) Worm: A worm is a self-replicating malicious program which uses a computer network to send copies of itself to other computers (nodes) on the network and it may do so without any user intervention. Unlike a virus, it does not need to attach itself to an existing program. Worms cause harm to the network by consuming bandwidth whereas viruses corrupt or modify files on a targeted computer.

iii) Trojan horse: A Trojan horse is malware that appears to perform a desirable function for the user prior to run or install but instead facilitates unauthorized access of the user's computer system. Once a Trojan horse has been installed on a target computer system, a hacker may have access to the computer remotely and perform various operations, limited by user privileges on the target computer system and the design of the Trojan horse.

iv) Spyware: Spyware is software that monitors a computer user's activity without their knowledge and reports it to a central location. The purpose of spyware ranges from purportedly benign (enforcing copyrights, displaying targeted advertisements) to very malicious (stealing passwords and credit card numbers).The most common way to get spyware on your computer is to install it yourself when you are tricked into installing free software.

v) Logic bomb: A logic bomb also known as slag code, is a piece of computer code that executes a malicious task such as clearing a hard drive or deleting specific files, when it is triggered by a specific event. It is secretly inserted into the code of a computer's existing software, where it lies dormant until that event occurs. This event may be a specific date and time or failure to input a command at a certain time.

vi) Software Key loggers: Software key loggers are software that record keystrokes entered by a user, usually to secretly monitor and/or maliciously use this information. They can record instant messages, email, passwords and any other information you type at any time using your keyboard. Software key loggers may also be embedded in spyware, allowing your information to be transmitted to an unknown third party over the Internet.

2.6 Measures to combat computer crime:

The following measures can be used to combat computer crimes:

i. Physical deterrents such as locks, card access keys, or biometric devices can be used to prevent criminals from gaining physical access to a machine on a network. Strong password protection both for access to a computer to a computer system and a computer's BIOS are effective countermeasures to against cybercriminals with physical access to a machine.
ii. Use access control mechanisms that will ensure confidentiality, integrity and availability.
iii. Encrypt confidential data stored in computers or transmitted over communication networks.
iv. Install antivirus software and update it regularly
v. Install intrusion detection systems to help detect any unauthorized access to the system.
vi. Install firewalls to prevent unauthorized access to local networks.
vii. Network vulnerability testing performed by technicians or automated programs can be used to test on a full-scale or targeted specifically to devices, systems, and passwords used on a network to assess their degree of secureness. Furthermore network monitoring tools can be used to detect intrusions or suspicious traffic on both large and small networks.
viii. Use bootable bastion host that executes a web browser in a known clean and secure operating environment. The host is devoid of any malware, where data is never stored on a device and the media cannot be overwritten. The kernel and programs are guaranteed at each boot. Some solutions have been used to create secure hardware browsers to protect users while accessing online banker.
ix. ***Regular backups and security*:** Just making something illegal or setting up regulations does not stop it happening. Responsible computer users need to take reasonable steps to keep their data safe. This includes regular backups and sufficient security with passwords.
x. ***Close down chat rooms*:** Some chat rooms have been closed down due to abuses, especially where children are vulnerable. Some have moderators who help to prevent abuses. Advice about sensible use is important; especially to never give personal contact details or arrange meetings without extreme caution.
xi. ***Reduce email spamming*:** This may be reduced by:
   - never replying to anonymous emails
   - setting filters on email accounts
   - reporting spammers to ISPs, who are beginning to get together to blacklist email abusers
   - governments passing laws to punish persistent spammers with heavy fines

2.7 Computer Systems Security

Computer system security is the process of preventing and detecting the unauthorized use of computer systems. *Prevention* helps stop unauthorized users from accessing any part of the

computer system by controlling access to the system, while *detection* helps determine whether or not someone attempted to break into the system, if they were successful, and what they may have done.
Computer security has three main goals, **confidentiality**, **integrity** and **availability**, which can be conveniently summarized by the acronym "**CIA**":

i) Confidentiality ensures that information is not accessed by unauthorized persons. In other words, it ensures that information is kept secret or private.
ii) Integrity ensures that information is not altered by unauthorized persons in a way that is not detectable by authorized users. That means that there is an external consistency in the system - everything is as it is expected to be.
iii) Availability ensures that the system is accessible and useable upon appropriate demand by authorized users. In other words, this means preventing denial-of-service.

Different mechanisms used to ensure the security of computer systems are **authentication**, **encryption**, **firewalls**, **digital signatures**, etc.

a) Authentication
Authentication is the process of determining if someone is who they declare to be. In simple terms, it is proving someone's identity. Authentication can be obtained by the user providing something they know (password), something they have (smart card) or something they are (biometrics).

i) Passwords: A password is a secret sequence of characters that is required to login to a system, thus preventing unauthorized persons from gaining access to the system. When authentication is done through the use of a password, knowledge of the password is assumed to guarantee that the user is authentic. Passwords can be guessed or cracked and so if anyone is using a password to protect their system, the following guidelines will help make it more secure:

- ✓ Don't choose an obvious password (like your name, date of birth or name of relative)
- ✓ Keep your password secret. Don't share it!
- ✓ Change your password regularly but not too often.
- ✓ Make your password at least eight characters long.
- ✓ Do not use common or proper words of phrases - these can be found using a dictionary cracker.
- ✓ Use a mixture of upper and lower case letters and numbers.

ii) **Smart Card**: A smart card is a small card that holds user authentication information. When the card is inserted into a card reader, electrical fingers wipe against the card. The information in the card is read and used to authenticate the person. Cards can be stolen and so are not as reliable as biometrics.

iii) **Biometrics**: Biometrics is the science and technology of measuring and analysing biological data. In computer security, it refers to the use of measurable biological characteristics such as

fingerprints, eye retinas, iris patterns, facial patterns, voice patterns, hand measurements and DNA, to identify a person. It is the safest authentication technique.

b) Encryption:

Encryption is the process of transforming a message using an algorithm into a form unreadable by anyone except the intended recipient. The original message is known as plaintext, the algorithm is cipher and the encrypted text is cipher text. To read an encrypted message, one must have access to a key that will enable them to decrypt it.
Encryption ciphers can be grouped into two: substitution and transposition ciphers.

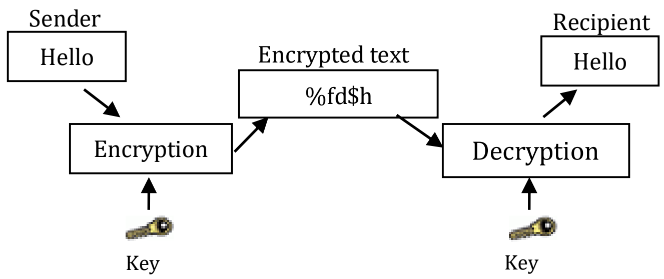


Figure 2.5: *encryption and decryption process in a transmission medium*

i) Substitution Ciphers

A substitution cipher is one in which the letters of the original message are replaced by other letters according to a key. Many substitution ciphers use only one alphabet, and are called mono-alphabetic ciphers. This means that we substitute one and only one letter for a particular letter in the message.
For example, every T in the message is replaced by the same substitute letter or symbol. Such a cipher scheme is easy to remember, but is also vulnerable to "cracking" using frequency analysis (letter counting).
In order to make substitution ciphers more secure, more than one alphabet can be used. Such ciphers are called polyalphabetic ciphers, meaning that the same letter of a message can be represented by different letters when encoded. Such a one-to-many correspondence makes the use of frequency analysis much more difficult in order to crack the code. Examples of substitution ciphers are Caesar cipher and Vigenere cipher.

a) The Caesar Cipher

The Caesar or shift cipher, named after Roman emperor Julius Caesar, is a mono-alphabetic substitution cipher in which each letter is translated into the letter a fixed number of positions after it in the alphabet table. A key number k is agreed upon by the sender and the receiver, then the standard alphabet is shifted k positions so that the k-th letter in the alphabet is substituted for letter A, the k+1st for B, etc. The alphabet is wrapped to maintain a one-to-one correspondence.
Example:
Suppose k = 3
Plain text letter: ABCD… WXYZ
Cipher text letter: DEFG…ZABC

Hence, the message: COMPUTING GIVES INSIGHT
Is translated into: FRPSXWLQJJLYHV LQVLJKW

b) The Vigenere Cipher

The Vigenere cipher is a polyalphabetic cipher based on using successively shifted alphabets, a different shifted alphabet for each of the 26 English letters. The procedure is based on the table below and the use of a keyword. The letters of the keyword determine the shifted alphabets used in the encoding process.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
B C D E F G H I J K L M N O P Q R S T U V W X Y Z A
C D E F G H I J K L M N O P Q R S T U V W X Y Z A B
D E F G H I J K L M N O P Q R S T U V W X Y Z A B C
E F G H I J K L M N O P Q R S T U V W X Y Z A B C D
F G H I J K L M N O P Q R S T U V W X Y Z A B C D E
G H I J K L M N O P Q R S T U V W X Y Z A B C D E F
H I J K L M N O P Q R S T U V W X Y Z A B C D E F G
I J K L M N O P Q R S T U V W X Y Z A B C D E F G H
J K L M N O P Q R S T U V W X Y Z A B C D E F G H I
K L M N O P Q R S T U V W X Y Z A B C D E F G H I J
L M N O P Q R S T U V W X Y Z A B C D E F G H I J K
M N O P Q R S T U V W X Y Z A B C D E F G H I J K L
N O P Q R S T U V W X Y Z A B C D E F G H I J K L M
O P Q R S T U V W X Y Z A B C D E F G H I J K L M N
P Q R S T U V W X Y Z A B C D E F G H I J K L M N O
Q R S T U V W X Y Z A B C D E F G H I J K L M N O P
R S T U V W X Y Z A B C D E F G H I J K L M N O P Q
S T U V W X Y Z A B C D E F G H I J K L M N O P Q R
T U V W X Y Z A B C D E F G H I J K L M N O P Q R S
U V W X Y Z A B C D E F G H I J K L M N O P Q R S T
V W X Y Z A B C D E F G H I J K L M N O P Q R S T U
W X Y Z A B C D E F G H I J K L M N O P Q R S T U V
X Y Z A B C D E F G H I J K L M N O P Q R S T U V W
Y Z A B C D E F G H I J K L M N O P Q R S T U V W X
Z A B C D E F G H I J K L M N O P Q R S T U V W X Y

Table 2.1: *The Vigenere tableau*

Example:

For the message COMPUTING GIVES INSIGHT and keyword TRUE, we proceed by repeating the keyword as many times as needed above the message, as follows.

T	R	U	E	T	R	U	E	T	R	U	E	T	R	U	E	T	R	U	E	T
C	O	M	P	U	T	I	N	G	G	I	V	E	S	I	N	S	I	G	H	T

The Vigenere tableau above can be used directly to encode the message. For each letter of the message use the letter of the keyword to determine a row and go across the row to the column headed by the corresponding letter of the message. The letter at the intersection is used to replace the letter of the message.

The above message is translated into: VFGTNACRZ XCZXJ CRLZALM

ii) Transposition Ciphers

A transposition cipher is one in which the letters of the original message are rearranged without otherwise changing them.

a) Columnar Transposition

A columnar transposition cipher enters the plaintext into a rectangle of a predetermined width and extracts cipher text by columns from left to right.

Example 1:
Plaintext: ENJOY THE BEAUTY OF SCIENCE

E	N	J	O	Y
T	H	E	B	E
A	U	T	Y	O
F	S	C	I	E
N	C	E		

Ciphertext: ETAFN NHUSC JETCE OBYI YEOE

Example 2:
Key: ORANGE
Plaintext: THE FUTURE IS BRIGHT

O	R	A	N	G	E
T	H	E	F	U	T
U	R	E	I	S	B
R	I	G	H	T	

Ciphertext: TUR HRI EEG FIN UST TB

b) Rail Fence Cipher

Rail fence writes the plaintext in a zig-zag pattern in two or more rows and forms the ciphertext by reading off the letters row by row from the first.

Example 1:
Plaintext: WE ARE HAVING FUN

			R						N				
		A		E				I		G			
	E				H		V				F		N
W						A						U	

Cipher text: RN AEIG EHVFN WAU

Example 2:
Cipher text: SL IIIFE HSAEC TRN

			S						L					
		I		I				I		F				E
	H				S		A				E		C	
T						R						N		

Plaintext: THIS IS RAIL FENCE

c) Firewall: A firewall is a system designed to prevent unauthorized access to or from a private network. Firewalls are implemented in either hardware or software form, or a combination of both. They prevent unauthorized Internet users from accessing private networks connected to the Internet. All messages entering or leaving the network must pass through the firewall which examines each message and blocks those that do not meet the specified security criteria. Some Operating Systems like Windows Vista, 7, 8 and Mac OS X, have built-in firewalls.

d) Intrusion Detection: Intrusion detection is the art and science of sensing when a system or network is being used inappropriately or without authorization. An intrusion-detection system (IDS) monitors system and network resources and activities and, using information gathered from these sources, notifies the authorities when it identifies a possible intrusion.

e) **Digital Signatures**: A digital signature (not to be confused with a digital certificate) is a mathematical technique used to validate the authenticity and integrity of a message, software or digital document.

The digital equivalent of a handwritten signature or stamped seal, but offering far more inherent security, a digital signature is intended to solve the problem of tampering and impersonation in digital communications. Digital signatures can provide the added assurances of evidence to origin, identity and status of an electronic document, transaction or message, as well as acknowledging informed consent by the signer.

2.8 Professional, Ethical and Moral Obligations of Users and Managers

a) Moral Obligations:

i) Contribute to society and human well-being: This principle concerning the quality of life of all people affirms an obligation to protect fundamental human rights and to respect the diversity of all cultures. An essential aim of computing professionals is to minimize negative consequences of computing systems, including threats to health and safety. When designing or implementing systems, computing professionals must attempt to ensure that the products of

their efforts will be used in socially responsible ways, will meet social needs, and will avoid harmful effects to health and welfare.

ii) Avoid harm to others: "Harm" means injury or negative consequences, such as undesirable loss of information, loss of property, property damage, or unwanted environmental impacts. This principle prohibits use of computing technology in ways that result in harm to any of the following: users, the general public, employees, and employers. Harmful actions include intentional destruction or modification of files and programs leading to serious loss of resources or unnecessary expenditure of human resources such as the time and effort required to purge systems of "computer viruses."

iii) Be honest and trustworthy: Honesty is an essential component of trust. Without trust an organization cannot function effectively. The honest computing professional will not make deliberately false or deceptive claims about a system or system design, but will instead provide full disclosure of all pertinent system limitations and problems. A computer professional has a duty to be honest about his or her own qualifications, and about any circumstances that might lead to conflicts of interest.

iv) Be fair and take action not to discriminate: The values of equality, tolerance, respect for others, and the principles of equal justice govern this imperative. Discrimination on the basis of race, sex, religion, age, disability, national origin, or other such factors is an explicit violation of ACM policy and will not be tolerated.

v) Honour property rights including copyrights and patent: Violation of copyrights, patents, trade secrets and the terms of license agreements is prohibited by law in most circumstances. Even when software is not so protected, such violations are contrary to professional behaviour. Copies of software should be made only with proper authorization. Unauthorized duplication of materials must not be condoned.

vi) Give proper credit for intellectual property: Computing professionals are obligated to protect the integrity of intellectual property. Specifically, one must not take credit for other's ideas or work, even in cases where the work has not been explicitly protected by copyright, patent, etc.

vii) Honour confidentiality: The principle of honesty extends to issues of confidentiality of information whenever one has made an explicit promise to honour confidentiality or, implicitly, when private information not directly related to the performance of one's duties becomes available. The ethical concern is to respect all obligations of confidentiality to employers, clients, and users unless discharged from such obligations by requirements of the law or other principles of this Code.

b) Professional Obligations:

i) Strive to achieve the highest quality, effectiveness and dignity in both the process and products of professional work: Excellence is perhaps the most important obligation of a professional. The computing professional must strive to achieve quality and to be cognizant of the serious negative consequences that may result from poor quality in a system.

ii) Acquire and maintain professional competence: Excellence depends on individuals who take responsibility for acquiring and maintaining professional competence. A professional must participate in setting standards for appropriate levels of competence, and strive to achieve those standards. Upgrading technical knowledge and competence can be achieved in several ways: doing independent study; attending seminars, conferences, or courses; and being involved in professional organizations.

iii) Know and respect existing laws pertaining to professional work: ACM members must obey existing local, state, province, national, and international laws unless there is a compelling ethical basis not to do so. Policies and procedures of the organizations in which one participates must also be obeyed. But compliance must be balanced with the recognition that sometimes existing laws and rules may be immoral or inappropriate and, therefore, must be challenged. Violation of a law or regulation may be ethical when that law or rule has inadequate moral basis or when it conflicts with another law judged to be more important. If one decides to violate a law or rule because it is viewed as unethical, or for any other reason, one must fully accept responsibility for one's actions and for the consequences.

iv) Accept and provide appropriate professional review: Quality professional work, especially in the computing profession, depends on professional reviewing and critiquing. Whenever appropriate, individual members should seek and utilize peer review as well as provide critical review of the work of others.

v) Give comprehensive and thorough evaluations of computer systems and their impacts, including analysis of possible risks:

Computer professionals must strive to be perceptive, thorough, and objective when evaluating, recommending, and presenting system descriptions and alternatives. Computer professionals are in a position of special trust, and therefore have a special responsibility to provide objective, credible evaluations to employers, clients, users, and the public. When providing evaluations the professional must also identify any relevant conflicts of interest, as stated in imperative 1.3

vi) Honour contracts, agreements, and assigned responsibilities.

Honouring one's commitments is a matter of integrity and honesty. For the computer professional this includes ensuring that system elements perform as intended. Also, when one contracts for work with another party, one has an obligation to keep that party properly informed about progress toward completing that work.

A computing professional has a responsibility to request a change in any assignment that he or she feels cannot be completed as defined. Only after serious consideration and with full disclosure of risks and concerns to the employer or client, should one accept the assignment. The major underlying principle here is the obligation to accept personal accountability for professional work. On some occasions other ethical principles may take greater priority.

A judgment that a specific assignment should not be performed may not be accepted. Having clearly identified one's concerns and reasons for that judgment, but failing to procure a change

in that assignment, one may yet be obligated, by contract or by law, to proceed as directed. The computing professional's ethical judgment should be the final guide in deciding whether or not to proceed. Regardless of the decision, one must accept the responsibility for the consequences.

However, performing assignments "against one's own judgment" does not relieve the professional of responsibility for any negative consequences.

v) Improve public understanding of computing and its consequences.

Computing professionals have a responsibility to share technical knowledge with the public by encouraging understanding of computing, including the impacts of computer systems and their limitations. This imperative implies an obligation to counter any false views related to computing.

c) Ethical Obligations:

i) Articulate social responsibilities of members of an organizational unit and encourage full acceptance of those responsibilities: Because organizations of all kinds have impacts on the public, they must accept responsibilities to society. Organizational procedures and attitudes oriented toward quality and the welfare of society will reduce harm to members of the public, thereby serving public interest and fulfilling social responsibility. Therefore, organizational leaders must encourage full participation in meeting social responsibilities as well as quality performance.

ii) Manage personnel and resources to design and build information systems that enhance the quality of working life: Organizational leaders are responsible for ensuring that computer systems enhance, not degrade, the quality of working life. When implementing a computer system, organizations must consider the personal and professional development, physical safety, and human dignity of all workers. Appropriate human-computer ergonomic standards should be considered in system design and in the workplace.

iii) Acknowledge and support proper and authorized uses of an organization's computing and communication resources: Because computer systems can become tools to harm as well as to benefit an organization, the leadership has the responsibility to clearly define appropriate and inappropriate uses of organizational computing resources. While the number and scope of such rules should be minimal, they should be fully enforced when established.

iv) Ensure that users and those who will be affected by a system have their needs clearly articulated during the assessment and design of requirements; later the system must be validated to meet requirements: Current system users, potential users and other persons whose lives may be affected by a system must have their needs assessed and incorporated in the statement of requirements. System validation should ensure compliance with those requirements.

v) Articulate and support policies that protect the dignity of users and others affected by a computing system: Designing or implementing systems that deliberately or inadvertently

demean individuals or groups is ethically unacceptable. Computer professionals who are in decision making positions should verify that systems are designed and implemented to protect personal privacy and enhance personal dignity.

vi) Create opportunities for members of the organization to learn the principles and limitations of computer systems: This complements the imperative on public understanding. Educational opportunities are essential to facilitate optimal participation of all organizational members. Opportunities must be available to all members to help them improve their knowledge and skills in computing, including courses that familiarize them with the consequences and limitations of particular types of systems. In particular, professionals must be made aware of the dangers of building systems around oversimplified models, the improbability of anticipating and designing for every possible operating condition, and other issues related to the complexity of this profession.

2.9 Legislation

Legislation (or "*statutory law*") is law which has been promulgated (or "*enacted*") by a legislature or other governing body or the process of making it.

i) The Computer Misuse Act (1990)

This Act makes it an offence to access any computer to which you do not have an authorized right to use. It introduces three criminal offences:

1. Accessing computer material without permissione.g looking at someone else's files.
2. Accessing computer material without permission with intent to commit further criminal offences, e.ghacking into the bank's computer and wanting to increase the amount in your account.
3. Altering computer data without permission, eg writing a virus to destroy someone else's data, or actually changing the money in an account.

ii) The Data Protection Act

The Act is aimed at protecting the rights of individuals to privacy. It protects personal data from being misused. Personal data is data that can identify an individual and allow an opinion to be expressed about them. Data such as a person’s name and address is not considered personal data but their date of birth and salary would be. The eight basic principles of the Data Protection Act are:

1. If an organization holds data on individuals, it must be registered under the act.
2. Personal data should be processed fairly and lawfully
3. Personal data should not be disclosed in anyway other than lawfully and within the registered purpose.
4. Personal data should be adequate and relevant and not excessive for the required purpose.
5. Personal data should be kept accurate and kept up to date

6. Data must be processed in accordance with the right of the data subject
7. Appropriate security measures must be taken against unauthorized access
8. Data should not be transferred to countries that do not have suitable data protection laws.

iii) Copyright, Design and Patent law

This Act is designed to protect all types of intellectual property and ensure that authors or creators of a piece of work receive both credit and compensation.

a) Copyright is a legal concept, giving the creator of original work exclusive rights to control its distribution for a certain time period. Something that is copyrighted is not to be reproduced, published or copied without permission from the copyright holder. Ideas are not protected by copyright; only the specific presentation of the idea is copyrightable.

Software licenses can be:
- Single user - licensed for installation on one computer
- Multi-user - the license allows you to install the software on a named number of computers
- Site-license - the license lets you install the software onto an unlimited number of computers, as long as they are on one distinct site such as a school

b) Design is the appearance or construction of something. A design is not immediately protected. It must be registered with the appropriate institution,

c) A **Patent** is a grant to inventors that give them exclusive monopoly over their invention. It gives them the right to stop others from producing, selling or using their invention. Unlike copyrights, patents protect the ideas or design of the invention rather than any tangible form of the invention.

iv) **Health and Safety Act**
The original Act and its many added regulations cover the range of hazards an employee may face like handling hazardous material. Some of the regulations that apply to the computing industry are:

a) **Display Screen Equipment Regulations**

They cover the precautions that must be taken when an employee uses a visual display unit. The regulation covers items such as the chair which must be adjustable, the desk which must be at the appropriate height, the monitor which must be adjustable and the lighting which must be appropriate.

b) **Moving and Handling Regulations**

These regulations lay down the rules for safe moving of heavy objects. All employees involved in such activities must receive proper training on avoiding injury when moving heavy objects.

c) **Control of Substances Hazardous to Health (COSHH)**

These regulations cover the safe storage and use of hazardous materials. This includes items such as laser printer toners and anyone involved in replacing such items must be made aware of the potentially toxic nature of toners.

2.10 Codes of Ethics and Professional Conduct

A code of ethics and professional conduct sets the standards for what is expected of a professional. They are promises by professions to regulate themselves in the general interest of the society. Code of ethics for information technology professionals encourage them to behave ethically and responsibly with the tools and information they have in their control. Examples are the British Computing Society (**BCS**) code of ethics, the Association for Computing Machinery (**ACM**) code of ethics, and the Institute of Electrical and Electronics Engineer (**IEEE**) code of ethics.

a) **ACM Code of Ethics**

1. General Moral Imperatives

An ACM member will…

1.1. Contribute to society and human well-being.
1.2. Avoid harm to others.
1.3. Be honest and trustworthy.
1.4. Be fair and take action not to discriminate.
1.5. Honour copyrights and patents.
1.6. Give proper credit for intellectual property.
1.7. Respect rights to limit access to computing and communication systems.
1.8. Respect the privacy of others.
1.9. Honour confidentiality.

2. More Specific Professional Responsibilities.

An ACM Computing Professional will . . .

2.1. Strive to achieve the highest quality, effectiveness and dignity in both the process and products of professional work.
2.2. Acquire and maintain professional competence.
2.3. Know and respect existing laws pertaining to professional work.
2.4. Accept and provide appropriate professional review.
2.5. Give comprehensive and thorough evaluations of computer systems and their impacts, including analysis of possible risks.
2.6. Honour contracts, agreements, and assigned responsibilities.
2.7. Improve public understanding of computing and its consequences.
2.8. Access computing and communications resources only when authorized to do so.

b) IEEE Code of Ethics

Members of the IEEE, in recognition of the importance of their technologies in affecting the quality of life throughout the world, and in accepting a personal obligation to their profession, its members and the communities they serve, commit themselves to the highest ethical and professional conduct and agree:

1. To accept responsibility in making decisions consistent with the safety, health and welfare of the public, and to disclose promptly factors that might endanger the public or the environment;
2. To avoid real or perceived conflicts of interest whenever possible, and to disclose them to affected parties when they do exist;
3. To be honest and realistic in stating claims or estimates based on available data;
4. To reject bribery in all its forms;
5. To improve the understanding of technology, its appropriate application, and potential consequences;
6. To maintain and improve our technical competence and to undertake technological tasks for others only if qualified by training or experience, or after full disclosure of pertinent limitations;
7. To seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, and to credit properly the contributions of others;
8. To treat fairly all persons regardless of such factors as race, religion, gender, disability, age, or national origin;
9. To avoid injuring others, their property, reputation, or employment by false or malicious action;

2.11 Netiquette

Netiquette is short for network etiquette. It is a set of rules about acceptable behaviour when communicating over the Internet. Some basic rules of netiquette are:

- i) Avoid flaming i.e. using obscene or inappropriate language in your emails or posts
- ii) Avoid using capital letters in your emails/comments, it is considered like YOU ARE SHOUTING and it is harder to read.
- iii) Avoid sloppiness i.e. avoid spelling and grammatical errors. Re-read and edit your emails/comments before you send/post
- iv) Do not send huge file attachments unless they are requested
- v) Always fill the subject field of an email before you send
- vi) Do not format your emails with colored text or background colour. They may cause them hard to read.

Drill Questions:

1. Describe the background to the increasing problems of data privacy, distinguish between data privacy and integrity of data?

2. Describe three ways where data privacy might be compromised.

3. Describe four ways in which integrity of data can be compromised. Suggest a solution to each problem.

4. Discuss the ways to secure a stand-alone computer. Why are the risks to privacy and integrity increased when a computer is networked? What steps can be taken to protect networked computers from attacks on privacy and integrity.

5. What do you understand by access control?

6. State what you understand by the following terms.
i) Digital inclusion ii) Digital divide.

7. (i) List and explain TWO main threats to the safety of data.
(ii) How does the Data Protection Act define the following terms?
(a) Personal data (b) Data (c) Data subject (**Q6(i,ii), CGCE2015**)

8.(i) Systems need protection from hazards such as flood and fire. Describe two other hazards that may impede a computer system. (**Q4(ii), CGCE2015**)
(ii) Explain three reasons why wireless communication is not preferred by some organisations.

9. Computers can be held responsible for a whole raft of health problems
(a) State and briefly explain two types of computer work related disorders
(b) Briefly explain how each can be prevented

10. Give two examples of computer crimes and two methods of preventing the stated crimes. (**Q1(ii), CGCE2014**)

11.(a) Define the term computer virus and computer anti-virus. (**Q6(iv), CGCE2014**)
(b) Briefly explain how the following malwares work: Trojan Horse, Worm, Boot sector Virus

12.(a) Explain the meaning of the term "Data Security".
(b) Describe how encryption will help to protect information or a message which is sent across a network. (**Q8(ii), CGCE2016**)

Suggested answers to some of the Questions:

2. *Unauthorized access (hacking), Accidental disclosure e.g leaving the screen open), deliberate disclosure e.g selling to journalists*

3. *Data may be corrupted :hardware failure; use e.g UPS and RAIDS and backup servers, a virus; anti-virus software, unauthorized access; security measures including ID, password, biometric devices, code pads, USB dongle, card reader, levels of permission(in line with 'pay grade'), malicious action or mistake; need good manuals and training and security procedures.*

5. *The limitation and control of access to a system through identification and authentication.*
Or *The process of limiting access to a system only to authorized users. This can be achieved through identification and authentication.*

6.(*i) Digital inclusion: Digital inclusion is commonly defined as the incorporation of information technologies into the community in order to promote education and improve the quality of life.*
In other words, it is the ability of individuals and groups to access and use information and communication technologies.

Digital inclusion is necessary as we move towards a technology based society to ensure that all individuals can participate fully in the economic, educational, civic and social activities of their community.

ii) Digital divide: The digital divide refers to the gap between those who have and those who lack access to computers and the Internet

Chapter 3: COMPUTER ORGANISATION AND ARCHITECTURE

3.1 Introduction

Computer architecture refers to those attributes of a system that have a direct impact on the logical execution of a program like the instruction set, the number of bits used to represent various data types, I/O mechanisms and techniques for addressing memory. On the other hand, computer organization refers to the operational units of the system and their interconnections that realize the architectural specifications like control signals, interfaces between computer and peripherals and the memory technology used.

So, for example, the fact that a multiply instruction is available is a computer architecture issue. How that multiply is implemented is a computer organization issue.

3.2 The Von Neumann Architecture

The Von Neumann architecture is a stored-program computer model that was designed by the Hungarian born Mathematician, John Von Neumann. It is based on three concepts:

i. Both data and instructions (programs) are stored in a single storage structure called memory
ii. The contents of this memory are addressable by location, without regard to the type of data contained there.
iii. It has a single processing unit which. As such, execution occurs in a sequential fashion from one instruction to the next.

Programs being stored in memory ensures that by altering the stored program, the computer can perform a different task – reason why a computer is called a general purpose machine.

Neumann divided the computer into four functional units: input, processing, storage and output.

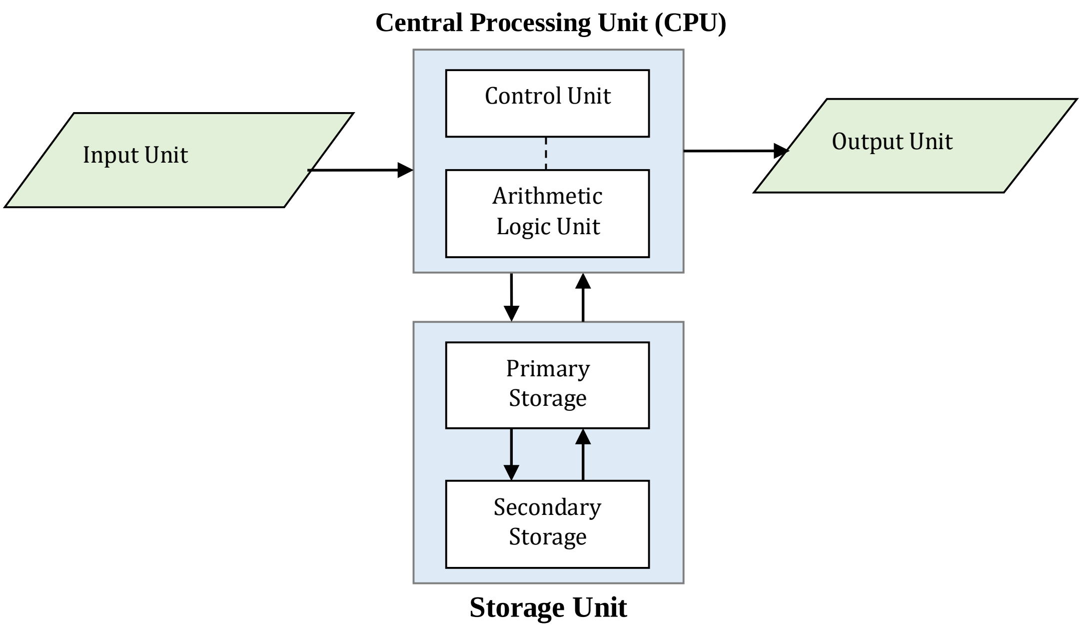


Figure 3.1: *Functional Units of a Computer*

3.2.1 Input Devices

Input devices are hardware components that are used to enter data and instructions into the computer. Examples are keyboard, mouse, scanner, joystick, light pen, touchpad, trackball and microphone.

i) Keyboard

The keyboard is the standard input device attached to computers. It has keys (buttons) that are pressed to enter data and commands into the computer. These keys represent letters, numbers, symbols and control characters. The keyboard has a total of 101-104 keys divided into different groups or keypads.

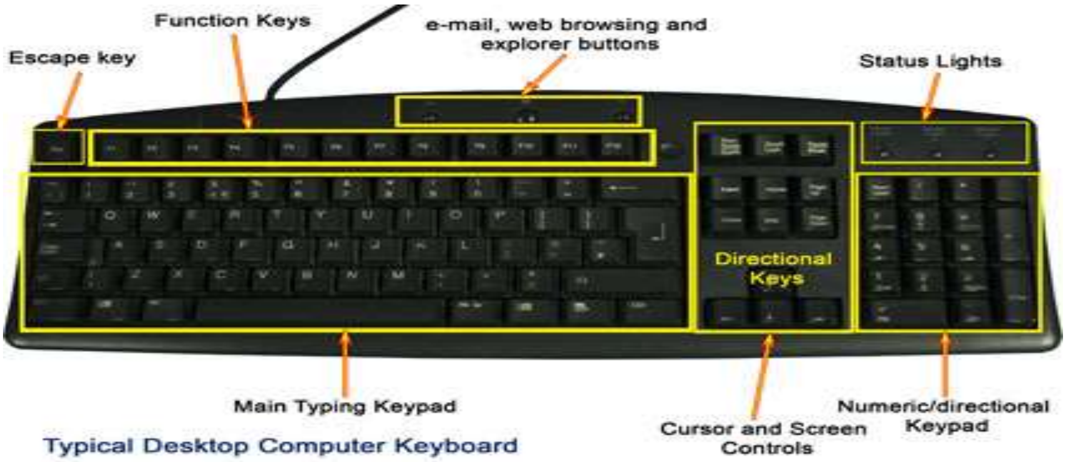


Figure 3.2: *Typical Desktop computer keyboard*

a) **Typing area**: It looks and arranged like a traditional typewriter where you press alphabetic keys. It holds alphabetic character such as letter, special characters and numbers. This is the area you use mostly when you do word processing.

b) **Function keys**: The functions keys are located at the top of a keyboard and grouped into four. There are 12 functions keys starting from F1 through F12. These keys are used for special purposes and most programmers use these keys to do a specific task. For example, if you are writing text with Microsoft Word and wanted to read Help, you can press F1 to display the Help. F5 key will display Find and Replace dialogue box. F12 key will display Save As dialogue box.

c) **Numeric keypad**: Numeric keypad is the other part of computer keyboard. Usually, it is located at the right side of a keyboard. It is arranged like a standard calculator used to enter numerical data. It can also be used as directional keys. Pressing the Num Lock key above the numeric keypad will tell whether the keys are on numeric or directional mode. If it is on, it is on numeric mode and can enter numbers. If it is off, it is on directional mode and only used for moving a cursor on screen UP, Down, Left or Right.

d) **Cursor and monitor controls**: These are keys found between the typing keypad and the numeric keypad. It has two groups of keys, arranged top and bottom. The top keys holds

e) **Insert**, **Home**, **Page Up**, **Page Down**, **Delete**, and **End** keys. Insert key switches between insert and overtype modes. Home key brings you back at the beginning of a page. Page Up and Page Down keys help you to move one page or screen up or down. Delete key erases a text or page. The End key takes you at the end of a page.

There are different keyboard layouts but the best known are *QWERTY* mainly used by English speakers and *AZERTY* used by French speakers. Another keyboard layout modified greatly from a standard layout is Dvorak, devised to increase typing speed by placing frequently used keys more naturally.
A keyboard connects to the computer through cable *PS/2* cable, USB cable or wireless (cordless).

ii) Mouse
A mouse is a handheld device which is moved across a flat surface to control the movement of a pointer on a computer screen. The mouse has two buttons, the right and the left buttons, which are clicked (pressed) to enter commands into the computer.
Different actions that can be performed with a mouse are:
- **Click**: Pressing the left mouse button.
- **Right-click**: pressing the right mouse button
- **Double-click**: pressing the left button two times in quick succession
- **Drag**: pressing the left button and while holding, moving the mouse
- **Drop**: Releasing the left button after dragging

Drag and drop action can be used to move files or documents from one place to another.
A mouse connects to a computer through PS/2 cable (PS/2 mouse), USB cable (USB mouse), or wireless (optical mouse).

iii) Scanner
A scanner is an optical device that converts hardcopy image or text into digital form so that it can be fed into the computer. The common optical scanner devices are the flatbed scanner, optical mark reader, optical character reader and barcode reader.
- **Flatbed Scanner**: a scanner in which the object to be scanned is held flat against a piece of glass.
- **Optical Mark Reader (OMR):** is the process of gathering information from human beings by recognizing marks on a document. OMR is accomplished by using a hardware device (scanner) that detects a reflection or limited light transmittance on or through piece of paper. OMR allows for the processing of hundreds or thousands of physical documents per hour. For example, students may recall taking tests or surveys where they filled in bubbles on paper with pencil. Once the form had been completed, a teacher or teacher's assistant would feed the cards into a system that grades or gathers information from them.
- **Optical Character Recognition:** Often abbreviated *OCR*, optical character recognition involves reading text from paper and translating the images into a form that the computer can manipulate (for example, into ASCII codes). An OCR system enables you to take a book or a magazine article, feed it directly into an electronic computer file, and then edit the file using a word processor.

- **Barcode Reader:** The machine-readable representation of the UPC (***U**niversal **P**roduct **C**ode*, a unique 12-digit number assigned to retail merchandise that identifies both the product and the vendor that sells the product.). Bar codes are read by a scanner that passes over the code and registers the UPC. The width of each black line and the subsequent white space between each line coincides with the numbers of the UPC

Other types of input devices are digital cameras, joystick, light-pen, tracker ball, touchpad, microphones, magnetic stripe readers and sensors.

3.2.2 Output Devices

Output devices are hardware components that are used to retrieve information from the computer. They provide the results of computations to the person using the computer, in a way they can understand. Examples are monitor, printer, speaker and projector.

i) Monitor

Also referred to as *VDU (Visual Display Unit),* the monitor is the most popular output device. It displays information generated by a computer on a screen. Such output is known as softcopy output. Monitors are characterized by the technology they use, their size and their resolution.

a) By the technology used, monitors are of two main types: Cathode Ray Tube (CRT) monitors and Flat Panel Display.
   - CRT monitors are similar to a television set.
   - Flat panel display monitors are of different types: Liquid Crystal Display (LCD), Electroluminescent Display (ELD), Gas Plasma (GP) and Thin Film Transistor (TFT) monitors.

b) The size of a monitor refers to how big the monitor is. The size is measured along the diagonal from the bottom left hand corner to the top right hand corner of the screen. Typical sizes are 10" or 12" for LCDs and 14", 15" or 21" for desktop monitors.

c) Monitor resolution refers to the number of dots (pixels) on the screen. It is expressed as a pair of numbers that give the number of dots on a row and the number of rows. A variety of different resolutions are available. For example, VGA is 640x480. This means that there are 640 pixels in each row across the screen and 480 pixels in each column up and down the screen. SVGA is 800x600.

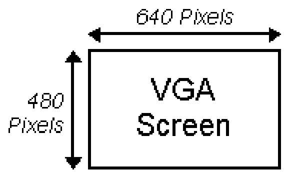


Figure 3.2: *Dimensions of a VGA screen*

The size of a pixel is known as dot pitch. The smaller the dot pitch, the higher the number of dots the screen has. Displays with lots of pixels are called high resolution while those with fewer pixels are called low resolution. The higher the resolution, the clearer and sharper the image will appear on the monitor. Most monitors come with a .28 dot pitch.

ii) **Printer**

A printer is a device that produces computer-generated information on paper. Such output is referred to as printout or hardcopy. Based on the technology used, printers can be classified into impact and non-impact printers.

1. Impact printers have mechanical contact between paper and printing head (e.g. daisy wheel, dot matrix and line printers).
2. Non-impact printers have no mechanical contact between paper and printing head (e.g. ink-jet, desk- jet, laser printers).

a) **Differences between Impact and Non-Impact computers**

1) **Descriptions**:

Impact printer - Produces text and images when tiny wire pins on print head strike the ink ribbon by physically contacting the paper

Non-impact printer - Produces text and graphics on paper without actually striking the paper

2) **Types**:

Impact printer - Dot-matrix printer

Non-impact printer - Inkjet printer, laser printer and thermal printer

3) **Speed**:

Impact printer - Low printing speed

Non-impact printer - Reasonably fast

4) **Quality**:

Impact printer - Print quality lower in some types

Non-impact printer - High quality of output, capable of printing fine and smooth details

5) **Letter quality**:

Impact printer - Produce near letter quality (NLQ) print only, which is just suitable for printing mailing labels, envelopes, or invoices

Non-impact printer–Letter-quality printouts

6) **Consumption**:

Impact printer - Not commonly used today

Non-impact printer - Most commonly used printer today

7) **Tools**:

Impact printer - Uses ink ribbon

Non-impact printer - Uses ink spray or toner powder

8) **Cost**:

Impact printer - Less expensive

Non-impact printer - More expensive

9) **Durability**:
Impact printer - Reliable, durable (lasting for a long time)
Non-impact printer - Print head is less durable, inclined towards to clogging and damage

10) **Sound effects**:
Impact printer - Generally noisy because of the striking activity
Non-impact printer - Generally much quieter than impact printers because there is no striking mechanism

11) **Image clarity**:
Impact printer - Poor graphics or none at all
Non-impact printer - Can handle graphics and often a wider variety of fonts than impact printers

12) **Multipart forms**:
Impact printer - Ideal for printing multipart forms because they can easily print through many layers of paper
Non-impact printer - Cannot print multipart forms

13) **Colour output**:
Impact printer - Limited colour printing
Non-impact printer - Capable of printing in strong clear colour, good for printing pictures

3.3 The Central Processing Unit

The central processing unit (CPU) also called processor, is the brain or heart of the computer. It is the part of the computer that interprets and executes program instructions. It also controls the other components of the system. The CPU is made up of three main components: the arithmetic-logic unit (ALU), the control unit (CU) and registers. Examples of CPUs are Intel Pentium II, III, IV, Pentium Celeron, and AMD Athlon.

3.3.1 Control Unit

The control unit acts like supervisor seeing that things are done as they ought to. It locates and retrieves program instructions from memory, interprets them and ensures that they are executed in proper sequence. It also ensures that data is selected from memory as necessary and information is stored correctly as well.

3.3.1 Arithmetic-Logic Unit

The arithmetic-logic unit (ALU) performs arithmetic and logic operations. It contains arithmetic circuits that perform arithmetic operations like addition, subtraction, multiplication and division, and logic circuits that perform comparisons like equal to, less than, greater than, greater than or equal to and less than or equal to.

3.3.2 Registers

Registers are special storage locations within the CPU that offer an advantage of speed. They work under the direction of the control unit to accept and hold data that is being processed.

Since the CPU uses registers for the processing of data, the number of registers in a CPU and the size of each register affect the power and speed of the CPU.
Registers are grouped into two: special purpose registers and general purpose registers.

- **Special purpose registers** are dedicated to specific tasks like:
  - the **accumulator** which collects the result of computations,
  - The **memory address register** (MAR) which keeps track of where a given instruction or piece of data is stored in memory
  - The **memory data register** (MDR) which holds data values.
  - The **program counter** (PC) which holds the address of the next instruction to be executed.
  - The **current instruction register** (CIR) which holds the instruction being executed.

- **General purpose registers** on the other hand have no specific function; they are used according to the need of the program being executed.

3.4 The Instruction Cycle

The instruction cycle describes how program instructions are repeatedly fetched, decoded and executed, one instruction at a time, until an instruction to HALT is encountered. Before an instruction can be fetched, it must be placed into memory as well as related data, from an input or secondary storage device. Once the necessary data and instructions are in memory, the central processing unit performs the following four steps for each instruction:

1. The control unit fetches (gets) data and instructions from memory.
2. The control unit decodes the instructions i.e. determines what they mean, and directs that the necessary data be moved to the arithmetic-logic unit.
3. The arithmetic-logic unit then executes the instruction on the data. That is, the ALU is given control and performs the actual operation on the data.
4. The arithmetic-logic unit stores the result of this operation in memory or in a register.

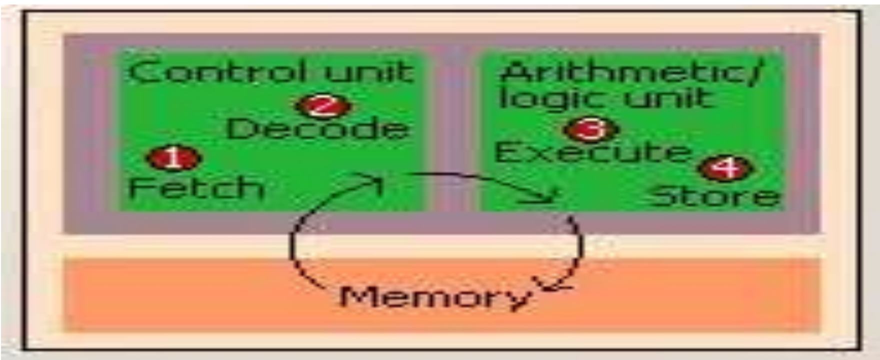


Figure 3.3: *stages Involved in the machine cycle*

Steps 1 and **2** together are called **instruction time** or I-time and **steps 3** and **4** together are called **execution time** or E-time. The combination of I-time and E-time is called the machine or instruction cycle or the fetch-decode-execute cycle. The length of time taken to fetch, decode and execute an instruction is measured in clock cycles.

3.5 System Clock

The CPU has a small quartz crystal circuit called the system clock that controls the timing of all computer operations. The system clock generates regular electronic pulses, or ticks, that set the operating pace of components of the system unit. Each tick is known as clock cycle and the pace of the system clock is called clock speed. Clock speed is measured in megahertz (Mhz) or gigahertz (Ghz) and refers to the number of clock cycles per second that the CPU runs at. Mega and giga stand for million and billion respectively while hertz means cycles per second. Thus, 1Ghz means one billion cycles per second. A computer that operates at 3 Ghz has 3 billion (giga) clock cycles in one second (hertz).
The faster the clock speed, the more instructions the processor can execute per second. The speed of the system clock has no effect on devices such as a printer or disk drive. The speed of the system clock is just one factor that influences a computer's performance. Other factors, such as the type of processor chip, amount of cache, memory access time, bus width, and bus clock speed.

3.6 Storage Devices

Storage devices are computer components that hold programs and data for use in the computer. Programs and data can be made available for initial or additional processing when required. A storage device is made of two parts: the storage medium and the device.

i. The medium is the surface or substrate that holds actual data
ii. The device reads information from or stores information onto the medium

Computer storage can be classified basically into two: *primary storage* and *secondary storage*.

3.6.1 Primary Storage

Primary storage is directly accessible to the CPU. It holds programs and data that the CPU is currently working with. Primary storage is also called internal memory, immediate access memory or primary memory. Primary memory consists of random access memory, read only memory and cache memory.

a) Random Access Memory

Random access memory (RAM) also called "main memory" is the temporary storage space into which a computer loads programs and user data when it is running. It is the computer's working space. It is read/write meaning that data can be read from and written onto it. RAM is also volatile meaning that everything held in it is lost when power is switched off. Two types of RAM exist: Static RAM and dynamic RAM.

i. **Dynamic RAM (DRAM)** consists of capacitors that slowly leak their charge over time. Thus, they must be refreshed every few milliseconds to prevent data loss. DRAM is cheap memory owing to its simple design.

ii. **Static RAM (SRAM)** consists of circuits that retain their charge over time. SRAM is faster and more expensive than dynamic RAM, and does not need to be refreshed as DRAM does. Due to its cost it is not used as main memory but rather to build cache memory.

b) Read Only Memory

Read only memory (ROM) is a kind of memory whose contents can only be read by the computer. Data found in ROM is written by the manufacturer and cannot be modified by the user. ROM is useful for holding data that never changes like the “boot” or start-up program which is run when the computer is switched on.ROM is non-volatile meaning that its content is preserved even without power. There are four types of ROM:

i. Masked ROM is ROM programmed with its data when the chip is fabricated.
ii. Programmable ROM (PROM) is ROM that can be programmed once but not reprogrammed.
iii. Erasable Programmable ROM (EPROM) is ROM that can be erased by strong ultraviolet light and new data burnt into it. To do this the chip has to be removed from the machine and put back after the changes have been made.
iv. Electrically Erasable Programmable ROM (EEPROM) is ROM whose content can be erased electrically. In this case, the chip need not be removed from the machine. The programming is done using special software.

c) Cache Memory

Cache memory is a smaller and faster memory between the CPU and main memory, which stores copies of data from the most frequently accessed memory locations. The purpose of cache memory is to speed up accesses by storing recently used data closer to the CPU, instead of storing it in main memory. Cache is static RAM and is usually organized in levels:

i. Level 1 (L1) cache, Level 2 and Level 3 cache.
ii. Level 1 cache is closest to the CPU or within it.
iii. L2 and L3 caches are outside it.

When the CPU needs to access memory, cache memory is examined first. If the data is found in cache, it is read from it. Otherwise, main memory is accessed. When the CPU refers to memory and finds the data in cache, it is said to be a HIT. Otherwise, it is a MISS.

3.6.2 Secondary Storage

Secondary storage is not directly accessible to the CPU. It is used to store programs and data for backup purposes (future use). It could be placed within the computer or connected externally. Programs and data from secondary storage must be transferred to main memory for processing. Secondary storage is also called secondary memory, mass storage, backing storage or external storage. It can be divided into magnetic storage, optical storage and solid state storage.

a) Magnetic Storage

Magnetic storage devices store data as electromagnetic charges on the magnetic surfaces of the storage units. Examples are floppy disks, hard disks and magnetic tape.

i. **Floppy Disk**

A floppy disk consists of a round flexible plastic disk coated with a magnetic substance and protected by a plastic cover lined with a soft material that wipes the disk clean as it spins. The

disk is made of two recordable surfaces which are divided into a number of circular paths called tracks. The number of tracks per surface varies with the particular type of disk. Each track is in turn divided into a number of smaller units called sectors. A sector is the basic unit of storage on the disk and has a capacity of 512 bytes.

ii. Hard Disk

A hard disk consists of several metallic platters which store data. Each platter has two sides and is divided into a number of rings called tracks. Tracks on a platter are numbered 0 from the outside and usually go up to 1023. Each track is divided into sectors. A sector is the basic unit of storage on the disk and has a capacity of 512 bytes. Sectors are grouped together to form clusters. A cluster is the smallest logical amount of disk space that can be allocated to hold a file. A cylinder is a sum set of all the tracks on all the platters that have the same track value.
Factors that determine the performance of hard disks are, seek time and drive rotational speed.

- Seek time is the time taken to move the read/write head over the right track and sector.
- Drive rotational speed is the total number of revolutions the disk platters make per minute. Higher rotational speed leads to higher transfer rate.

iii. Magnetic Tape

A magnetic tape consists of a magnetically coated stripe on which data is stored. Data is stored on the magnetic tape in chronological order or sequentially. This means that any piece of data is always stored in the next available space on the tape. To access data, the tape drive has to move through all the preceding data before it can access the desired data. This mode of access is known as sequential access. Less susceptible to environment, they are suitable for long-term storage and backup.

b) Optical Storage

Optical storage devices store data as microscopic light and dark spots on the disk surface. Examples are Compact discs, digital versatile discs and Blu-ray discs. They are less susceptible to environmental damage.

i. Compact Disc (CD)

A CD is a round disk coated with a metallic surface on which data can be stored and accessed via laser technology. To store or access data on a CD, the CD drive focuses a laser beam on the disc surface. Different variations of CD exist: CD-ROM, CD-R and CD-RW. A CD can store 650MB to 700MB of data.

- CD-ROM stands for compact disc read only memory. CD-ROMs can only be read but not recorded on, by the user's computer. Their content is set during manufacture.
- CD-R stands for compact disc recordable. It is a type of CD that can be recorded by the user. Once the user records on the CD, the content is set and cannot be changed. CD-R can be read by CD-ROM drives but to write on them, you need a CD-R drive.
- CD-RW stands for compact disc rewritable. It is a type of CD that can be recorded, erased and reused by the user. CD-RW cannot be read by a CD-ROM and CD-R drives. CD-RW drives are required to read and write on them.

ii. Digital Versatile Disc (DVD)

A DVD is similar to a CD in size and thickness but has a higher storage capacity than the CD. DVDs use a laser beam of wavelength shorter than used by CDs. This allows for smaller indentations and increased storage capacity. Just like the CD, different ions of the DVD exist: DVD-ROM, DVD-R and DVD-RW. A DVD can store up to 17GB of data. Common DVD storage capacities are:

Types	Characteristics	Capacity
DVD-5	Single-sided, Single-layer	4.7GB
DVD-9	Single-side, Dual-layer	8.5GB
DVD-10	Double-side, Single-layer	9.4GB
DVD-18	Double-side, Dual-layer	17.1GB

iii. Blu-Ray Disc

A Blu-ray disc is an optical disc similar to a DVD and of same size, but read and written with a blue or violet laser, whose shorter wavelength makes a higher data density possible. Blu-ray discs can hold 25 GB for single layer or 50 GB for double layer.

iv. USB Flash Drive

A flash drive is a small, keychain-sized flash memory device with a USB interface, treated by the computer as if it were a disk drive. A flash drive is also called thumb drive, jump drive or memory stick.USB flash drives have practically replaced diskettes as a handy way to transport data. They can be carried in one's pocket and plugged into any computer for immediate access.

v. Secure Digital Cards

A secure digital (SD) card is a type of flash-memory card that incorporates a cryptographic security system to prevent copyright violations, often used in digital music players and digital cameras.

vi. Solid State Storage

The term solid-state essentially means no moving parts. Solid-state storage devices are based on electronic circuits with no moving parts (no reels, no spinning disks). They store data using a special type of memory called flash memory. Flash memory is a type of EEPROM that can only be erased in blocks; it cannot be erased one byte at a time. In this regard it resembles a disk that is divided into sectors. Examples of solid state devices are USB memory sticks, memory cards and secure digital cards.

3.6.3 Characteristics of Storage Devices

Four important characteristics of storage devices are capacity, access time, access method and volatility.

a) Capacity

The capacity of a storage device is the maximum amount of data that can be stored on the device's medium. It is expressed in terms of the number of data bytes the device can hold. This

simply means the unit of measurement of storage capacity is byte. Units of storage are summarized as follows:

1 bit = 0 or 1	
4 bits = 1 nibble	
8 bits = 1 byte	1 character (letter, number or symbol)
1024 bytes = 1 kilobyte (KB)	approx. 1/2 page
1024 KB = 1 megabyte (MB)	approx. 500,000 pages
1024 MB = 1 gigabyte (GB)	approx. 5 million pages
1024= 1 terabyte (TB)	approx. 5 billion pages

b) Access time

Access time is the average time taken for a storage device to search and read required data on its medium. In other words, it is how fast data can be read from or written to a memory device's medium. It is measured in seconds. Units of time are summarized as follows:

c) Access Method

An access method is the technique used to retrieve information from or store information to a medium. Storage media can be accessed in two ways: sequentially or randomly.

i. Sequential Access

The medium is accessed by proceeding from the beginning of the medium until the designated area is reached. Any new data is stored in the next available space on the medium. To read any data stored on the medium, the device has to start from the beginning going through each data until the required data is found. An example of sequential access medium is magnetic tape.

ii. Random Access

Data is accessed in any order, regardless of its location on the medium. To read any data stored on the medium, the device does not need to go through all preceding data. It is also called direct access. Examples of direct access devices are RAM, ROM, CDs, DVDs and magnetic disks.

d) Volatility

Volatility refers to the behaviour of the device without power. A device can be volatile or non-volatile. Volatile means that the device loses its content when power is switched off. Examples of volatile devices are Cache and RAM.

Non-volatile means that the device preserves its content even without power. Examples of non-volatile devices are ROM, magnetic disks, optical discs and solid state devices.

3.6.4 Memory Hierarchy

Modern computers manage memory by organizing memory into a hierarchy in which large and slower memories feed data into smaller but faster memories for faster processing of data. This organization of computer memory is known as memory hierarchy. At the top of the hierarchy are the CPU registers followed by cache memory. The next level in the hierarchy is made up of main memory, which is followed by magnetic disk.

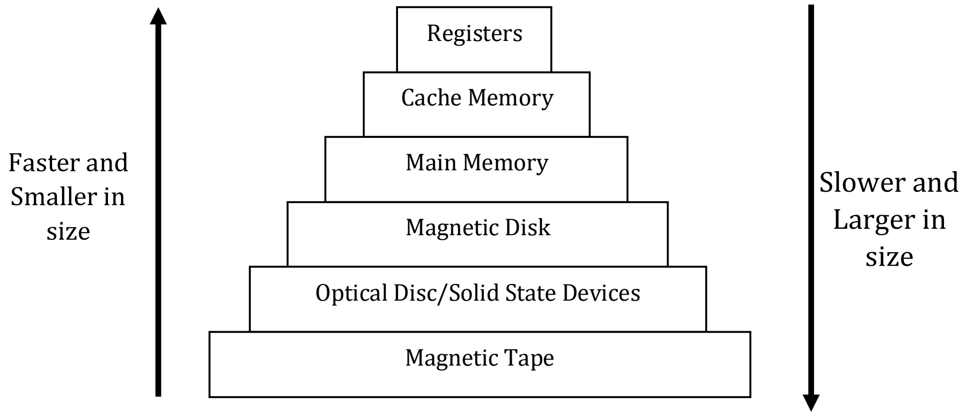


Figure 3.4: *Memory hierarchy*

3.7 The Motherboard

The motherboard is the main printed circuit board in the computer which holds the chipset and other electronic components that give function to the computer. The motherboard is indispensable to the computer and provides the main computing capability.

3.8 Motherboard Form Factor

Motherboard types are better described by what we call the motherboard form factor**.** The form factor of a motherboard determines the specifications for its general shape and size. It also specifies what type of case and power supply will be supported, the placement of mounting holes, and the physical layout and organization of the board. The most common form factors found in modern PCs are:

i. Advanced Technology (AT)
ii. Advanced Technology Extended (ATX)
iii. Low Profile Extension (LPX)
iv. New Low profile Extended (NLPX)

3.9 The System Bus

A computer bus is a set of parallel lines that interconnects various components inside the computer, allowing the exchange of data between them. It is the pathway between these components, enabling data to be transferred from one component to another. The width or size of a bus is determined by the number of lines it has. The system bus is made up of three different busses: the data bus, the address bus and the control bus.

a) The Data Bus

The data bus carries data between the CPU and main memory or peripherals. During a write operation, data is carried from the CPU and during a read operation, data is carried into the CPU. This means that the data bus is bidirectional. The size of the data bus determines how much data can be transferred in a single operation.

b) The Address Bus

The address bus carries address information from the CPU to main memory or peripherals. It is unidirectional. The CPU uses the address bus to send the address of the memory location to be written to or read from. Also, when the CPU reads data from or writes to a port, it sends the port address out on the address bus. The size of the address bus determines the maximum

amount of main memory (RAM) that can be addressed. A computer with a 32 bit bus size can address a maximum RAM of 2^{32} bits = 4GB.

c) The Control Bus
The control bus is used by the CPU to send out signals to enable the outputs of addressed memory devices or port devices. Typical control bus signals are memory read, memory write, I/O read and I/O write.

When the CPU wants to read data from a memory location, it sends out the memory address of the desired data on the address bus and then sends out a Memory Read signal on the control bus. The memory read signal enables the addressed memory device to output the data onto the data bus. The data from the memory travels along the data bus to the CPU.

3.10 Input/output Interfaces

The CPU communicates with I/O devices through bus interfaces connected to the system bus. These bus interfaces also called expansion buses are then connected to ports which allow the exchange of data and information between the computer and external (peripheral) devices.

a) I/O Ports
A port is a pathway for data and information to go into and out of the computer from external devices such as keyboards, monitors and printers.
There are many standard ports designed for special purposes. Examples are:

i. **PS/2 port** for connecting old computer mouse and keyboard. Most of the old computers provided two PS/2 ports, one for the mouse and the other for the keyboard
ii. **VGA (Video Graphics Adapter) port** for connecting the monitor to a computer's video card.
iii. **RJ45 (Ethernet) port** for connecting an Ethernet network cable to a computer.
iv. **RJ14 (Modem) port** for connecting a PC's modem to a telephone line
v. **USB (Universal Serial Bus) port** for connecting all kinds of external devices like flash drives, printers, keyboards, mice, and external hard drives. USB compliant devices can get power from a USB port.
vi. **Serial ports** for connecting serial devices like PDAs and external modems.
vii. **Parallel port** connecting parallel devices like printers and scanners
viii. **FireWire port** for connecting camcorders and video equipments to the computer. Data travels at 400 to 800 megabits per seconds. It was invented by apple.
ix. **DVI (Digital Video Interface)** port for connecting flat panel LCD monitors to the computer's high end video graphic cards.
x. **Sockets** for connecting microphones and speakers to a computers sound card

b) I/O Bus Standards
Ports follow standards that define their use. Examples of such standards are:

1. **SCSI**: Small Computer System Interface
2. **USB**: Universal Serial Bus
3. **IDE**: Integrated Device Equipment
4. **PCI**: Peripheral Component Interconnect

5. **ISA**: Industry Standard Architecture
6. **EISA**: Extended ISA
7. **VESA**: Video Electronics Standard Architecture
8. **SIMM**: Single Inline Memory Module
9. **DIMM**: Dual Inline Memory Module
10. **PCMCIA**: Personal Computer Memory Card International Association

3.11 Accessing Input/output Devices

External devices communicate with the system using a shared bus system. Input/output operations can be performed in three basic techniques: polling, interrupts and direct memory access.

a. Polling

Polling is a technique whereby the CPU constantly looks to see if a device needs its attention. In a polling system, the processor repeatedly checks a status flag to achieve the required synchronization between the processor and an input or output device. We say that the processor *polls* the devices. When an I/O device wants to transmit data, it sets the status flag.

b. Interrupts

In an interrupt driven interface, synchronization is achieved by having the I/O device send a special signal, called interrupt request or simply interrupt, over the bus whenever it is ready for a data transfer operation.

Definition*: An interrupt is a signal generated by hardware or software that causes the CPU to suspend what it is doing to handle another task of higher priority*.

Any event that will cause an interrupt is called an interrupt request (IR). For example, a key pressed on the keyboard. An interrupt handler or interrupt service routine (ISR) is a program that services an interrupt request. It contains the actions that will be executed for a given interrupt request.

c. Direct Memory Access

Instead of interrupting the CPU, direct memory access (DMA) can be use to relieve the CPU from the burden of I/O. With DMA, the device can transfer directly to memory without informing the CPU. To do this, a special control unit called DMA controller is needed. The DMA controller performs functions that would normally be carried out by the processor. The DMA controller must increment the memory address and keep track of the number of transfers. The operations of DMA controller must be under the control of a program executed by the processor. To initiate the transfer of block of words, the processor sends the starting address, the number of words in the block and the direction of the transfer. On receiving this information, DMA controller transfers the entire block and informs the processor by raising an interrupt signal. While a DMA transfer is taking place, the processor can be used to execute another program. After the DMA transfer is completed, the processor can return to the program that requested the transfer.

3.12 Flynn's Taxonomy of Computer Architecture

Michael Flynn classified computer architecture into four distinct categories based on the notion of stream or flow of information into the CPU. He identified two types of information streams:

1. The instruction stream which is the flow of instructions into the processing unit
2. The data stream which is the flow of the data on which the instructions are performed.

3.12.1 Single Instruction, Single Data Stream

A SISD machine is a uniprocessor machine which receives a single stream of instructions that operate on a single stream of data. At any given moment, a single instruction is being executed on a given data set. In SISD machines, instructions are executed sequentially. That is, instructions are executed one after the other, one at a time. Hence these machines are also called sequential or serial processor machines. They are typical examples of Von Neumann's computer model.

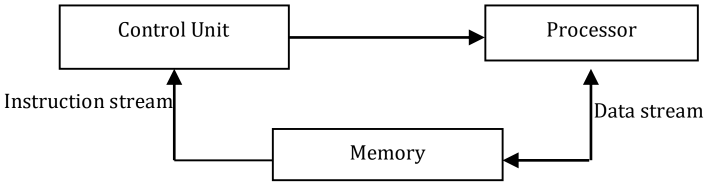


Figure 3.5: *SISD*

3.12.2 Single Instruction, Multiple Data Stream

SIMD machines are multiprocessor machines which are capable of executing the same instruction on all the processors at the same time. They have *n* identical processors which operate under the control of a single instruction stream issued by a central control unit on different data sets. At any given moment, the control unit broadcasts the same instruction to all processors which operate on different data sets from distinct data streams. SIMD machines are also called array processor machines. An example is the CRAY's vector processing machine.

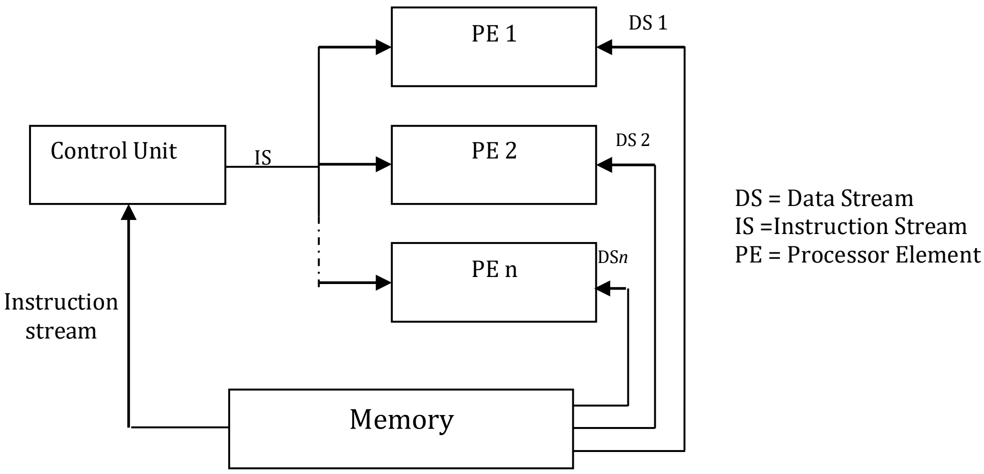


Figure 3.6: *SIMD*

3.12.3 Multiple Data, Single Instruction Stream

MISD machines are multiprocessor machines capable of executing different instructions on different processors, but all of them operating on the same data set. They have *n* processors, each with its own control unit, that share a common memory. Each processor receives a distinct instruction stream but all operate on the same data stream. Such machines no longer exist.

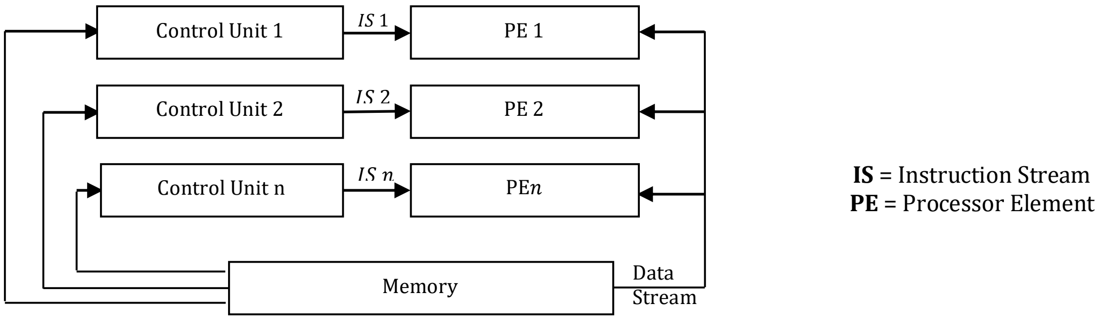


Figure 3.7: *MISD*

3.12.4 Multiple Data, Multiple Instruction Stream

MIMD machines are multiprocessor machines capable of executing multiple instructions on multiple data sets. They have *n* processors, *n* streams of instructions and *n* streams of data. Each processor element in this model has a separate instruction stream and data stream hence such machines are well suited for any kind of application. Examples of MIMD machines include multi-core computers like the Blue Gene, Fujitsu K computer and CRAY Jaguar.

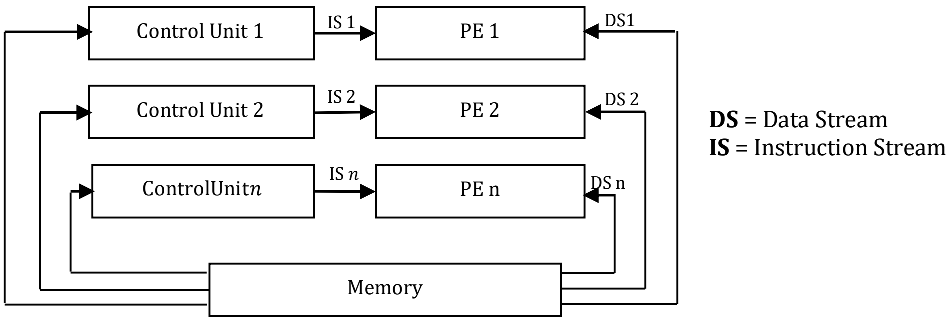


Figure 3.8: *MIMD Architecture*

3.13 Parallel ism

The performance of computers can be increased by making them perform a number of operations in parallel. This is known as parallelism. Parallelism can be achieved through parallel processing and pipelining.

3.13.1 Parallel Processing

Parallel processing is the use of multiple (independent) processors simultaneously to execute a single program or task. The problem is divided into portions so that multiple processors work on their assigned portion of the problem at the same time. Parallel processing requires special software that recognizes how to divide the problem and then bring the results back together again.

Some personal computers implement parallel processing with multiprocessors while others have multicore processors.

a. Multicore Processors

Multiple processing units can be fabricated on a single chip. In this case, each of these processors is termed **core**. The term processor is then used for the complete chip. Hence, we have the terminology ***dual-core***, ***quad-core***, and ***octo-core*** processors for chips that have two, four, and eight cores, respectively.

b. Multiprocessors

Computer systems may contain many processors, each possibly containing multiple cores. Such systems are called *multiprocessors*. These systems either execute a number of different application tasks in parallel, or they execute subtasks of a single large task in parallel. All processors usually have access to all of the memory in such systems, and the term *shared-memory multiprocessor* is often used to make this clear. The high performance of these systems comes with much higher complexity and cost, arising from the use of multiple processors and memory units, along with more complex interconnection networks.

3.13.2 Pipelining

The simplest way to execute a sequence of instructions in a processor is to complete all steps of the current instruction before starting the steps of the next instruction. If we overlap the execution of the steps of successive instructions, total execution time will be reduced.
For example, the next instruction could be fetched from memory at the same time that an Arithmetic (execute) operation is being performed on the register operands of the current instruction. This form of parallel processing is called *pipelining*.

3.14 Processor Architecture

A processor's architecture refers to the way in which its memory and control are organized.

3.14.1 Instruction Set Architecture

An instruction set is the collection of bit patterns or binary codes for the machine operations that a processor has been designed to perform. One of the most important characteristics that distinguish different computers is the nature of their instructions. There are two fundamentally different approaches in the design of instruction sets for modern computers CISC and RISC.

3.14.2 Complex Instruction Set Computer (CISC)

CISC (pronounced *sisk*) is a CPU design with a large amount of different and complex instructions. In CISC processors, the control unit contains a number of micro-electronic circuitry to generate a set of control signals and each micro-circuitry is activated by a microcode. Complex instructions are first decoded and the corresponding microcode routine dispatched to the execution unit.
The standard features of CISC processors are:

i. A large number of different and complex instructions.
ii. The use of complex addressing modes.
iii. Execution of different machine programs on a CISC machine.
iv. The use micro-program control unit.
v. Limited number of registers.
vi. Variable length instruction encoding
vii. Direct memory access

Examples of CISC processors are: Intel 386, 486, Pentium, Pentium Pro, Pentium II, Pentium III, Motorola's 68000, 68020, 68040, etc.

3.14.3 Reduced Instruction Set Computers (RSIC)

RISC (pronounced *risk*) is a CPU design with a small number of basic and simple machine language instructions, from which more complex instructions can be composed. RISC instructions are hardwired. That is, they are built into the chip with hardware rather than programming. Hardware implementation of instructions is much faster and uses less silicon than a microcode implementation.

The standard features of RISC processors are:

i. A small and limited number of instructions.
ii. The use of hardwired control unit.
iii. Consumption of less power and high performance
iv. Instructions are very simple and consistent
v. The use of simple addressing modes
vi. Fixed length instructions (easier to decode)
vii. Pipelining possible because fixed length instructions

Examples of RISC processors are: IBM RS6000, MC88100
DEC's Alpha 21064, 21164 and 21264 processors, Motorola/IBM PowerPC

3.15 Addressing Modes

A machine instruction specifies to the CPU what to do, where the data is located, and where the output data (if any) will be put. Every instruction is made up of two parts: Op-code and operand.

- ✓ The Op-code (operation code) denotes the basic machine operation e.g. ADD, STORE, SHIFT, XOR.
- ✓ The operand (one or more) provides the data which the instruction manipulates.

For example,

`ADD A, $0E` ADD is the op-code; A and $0E are operands

`MOV AX, #0` MOV is the op-code; AX and #0 are operands

Operands can be specified in different ways in an instruction. The way an operand is specified in an instruction is known as an addressing mode.

Definition*: An addressing mode refers to the way in which the processor locates the data (operand) associated with an instruction.*

a) Register Addressing

In register mode, operand is the contents of a processor register; the name of the register is given in the instruction. For example:

```
ADD R1, R2, R3
```

The above instruction uses three register to hold all operands. Registers R2 and R3 hold the two source operands while register R1 holds the result of the computation.

b) Immediate Addressing

In immediate mode, the operand is given explicitly in the instruction. In other words, the value to be manipulated by the instruction is immediately part of it. There is no need for any additional information. For example:

```
ADD R2, R3, #10
```

This instruction adds the number 10 to the content of register R3 and stores the result in register R2. The operand 10 has been addressed immediately.

c) Direct Addressing

In this addressing mode, the operand is in a memory location; the address of this location is given explicitly in the instruction. It is also called absolute addressing. For example:

```
LOAD R2, $0FF
```

The above instruction loads the content of the memory location $0FF into register R2. The operand $00FF has been addressed directly.

d) Indirect Addressing

Here, the effective address of the operand is the contents of a register that is specified in the instruction. For example:

ADD R2, R2, #R3

This instruction adds the contents of the memory location held in register R3 to the content of register R2 and the result of the operation is held in register R2.

e) Indexed Addressing

The effective address of the operand is generated by adding a constant value to the contents of a register.

Drill Questions

1. What is the difference between a computer with a Von Neumann's architecture and one with a Harvard architecture?
2. State the advantages of flat panel monitors over CRT.
3. State the advantages and disadvantages of impact and non-impact printers
4. How would you define processing?

Answer: *Processing is performing arithmetic operations (add, subtract, multiply, divide etc) or logical operations (comparisons like less than, greater than, equal to etc) on data to convert it to useful information.*

5. SIMD, MISD and MIMD machines are known as parallel processing machines. What do you understand by parallel processing?
6. What is pipelining (pipeline processing)?

7.(a) Explain one stage of the machine instruction cycle that takes place in the ALU of the CPU.
(b) What is Memory and describe the pourpose of ROM in the computer system.
(c) Explain the effect and the consequneces on the computer system in the absence of each of the following; (a) RAM (b) Buffers (c) NIC
(d) Given that a disk has 18 sectors, 80tracks and a sector capacity of 512bytes. Prove that the storage capacity of the disk is 1.44MB. (**Q4i,ii;7iv, CGCE 2016**)

9.(i)(a) Draw a block diagram that shows how input, process, main memory, output and storage relate in a computer.
(b) select any two blocks of you choice from (**a**) and name a device each.
(ii) Using the block diagram give the definition of a computer.
(iii) State the two blocks involved in the machine cycle.
(iv) State the four stages of the machine cycle.
(v) Explain the term volatile memory and give an example. (**Q5, CGCE 2014**)
10.(i)(a) Explain briefly the functions of TWO main components of a CPU.
(b) What is a cache memory?
(ii) Sketch a block diagram of a computer system that is based on the Von Newman machine architecture and identify its components.
(iii) How do modern PC systems implement Von Newman machine so that memory can be used more efficiently? (**CSC Q8i;9iii, CGCE 2013**)

Chapter 4: SOFTWARE

4.1 Introduction

A computer system is basically made up of hardware and software. Hardware refers to the physical and tangible components of the computer like the keyboard, mouse, monitor and internal circuits. Software on the other hand, refers to the collection of computer programs and data that run on a computer, and which make the hardware useful. They are the intangible components of the computer system.
Computer software has two major categories namely system software and application software.

4.2 System Software

System software control and coordinate computer resources (hardware and operations) so that the computer user and applications can smoothly interact. They help the computer carry out its basic operating tasks. System software are designed to perform computer related tasks. They include operating systems, firmware, utility programs, device drivers, library programs and language translators.

i) Basic Input Output System
Basic input output system (BIOS) is software that contains hundreds of programs that allow for communication between the CPU and devices. It is stored on ROM, which is a permanent chip on the motherboard.
There are three kinds of BIOS for hardware devices (*in this section we are just going to mention two kinds*):

1. Permanent never changing BIOS for never changing hardware like the keyboard
2. BIOS for hardware that changes occasionally. It requires extra volatile information so it is stored on a separate chip called CMOS (**C**omplementary **M**etal **O**xide **S**emiconductor).

ii) **Utility Software**
Utility software is used to enhance the operating system, or in some other way improve the usefulness of the system. They help analyse, configure, optimize and maintain the computer. Rather than providing user-oriented or output-oriented functionality, utility software focuses on how the computer infrastructure operates. Most major operating systems come with several pre-installed utilities. Examples of utility software include: disk defragmenters, backup utilities, disk compression utilities, disk cleaners, file managers, disk formatters and virus checkers.

1. **Disk defragmenters** detect computer files whose contents are broken across several locations on a disk, and move the fragments to one location to increase efficiency.
2. **Disk cleaners** find and delete files that are unnecessary to computer operation, or take up considerable amounts of space. They help users decide what to delete when their hard disk is full.
3. **Backup utilities** make copies of all information stored on a disk, and restore either the entire disk (e.g. in an event of disk failure) or selected files (e.g. in an event of accidental deletion).
4. **Disk compression utilities** reduce the space that a file takes up on disk, increasing the capacity of the disk.

5. **File managers** provide a convenient method of performing routine data management tasks, such as deleting, renaming, moving, copying, merging, generating and modifying files.
6. **Disk partition utilities** divide an individual drive into multiple logical drives, each with its own file system which can be mounted by the operating system and treated as an individual drive.
7. **Disk formatters**
8. **Virus checkers** prevent, detect, and remove malware.

4.3 Device Driver

A device driver is software that allows interaction between the operating system and a hardware device. It is an interface for communicating with the device through the specific computer bus that the hardware is connected to. Without an appropriate device driver the system cannot communicate with a device, rendering the device useless. Installation of device drivers usually happens automatically when hardware is connected (plug n play), or from a CD provided with the device. Sometimes a device driver needs to be updated to stay functional.

4.4 Language Translators

A language translator is a computer program that translates program instructions from one programming language to another. There are three types of language translators: compilers, interpreters and assemblers.

i. Compilers translate instructions written in a high-level language into machine language instructions.
ii. Interpreters translate instructions written in a high-level language into machine language instructions and execute them, one line at a time.

Assemblers translate instructions written in assembly language to machine language instructions.

4.5 The Operating System

The operating system is the essential software that is required for a computer to become operational. It is the software layer that is on top of the hardware to provide functionality to computer components, manage the hardware and serve as interface between the computer user and the computer. The operating system is called a virtual machine as it hides the complexity of the hardware and presents the user with an interface that is easier to understand and program. The operating system is stored on disk, but it needs to be loaded into memory (RAM) once the computer is switched on and before any other program can be run. The term bootstrapping refers to the process of loading the operating system into a computer's memory. This process is done by a program called the bootstrap loader that is stored permanently in the computer's electronic ROM chip. Examples of operating systems are Windows (95, 98, 2000, XP, Vista, 7 and 8), Macintosh Operating System (Mac OS), Linux and UNIX.

4.5.1 Operating System Structure

The operating system can be divided into two main components: kernel and command interpreter (shell).

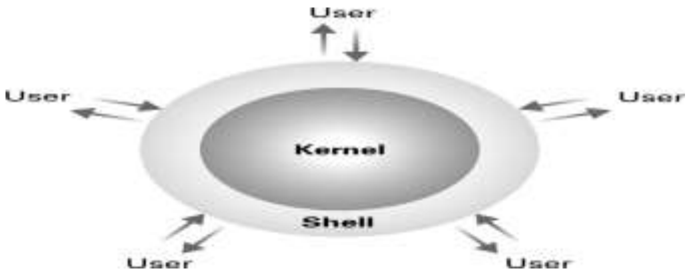


Figure 4.1: Structure of the Operating System

a) OS Kernel

The kernel is the central part of an operating system that is running at all times on the computer. It loads first and remains in memory as long as the computer is on. It consists of utilities (file manager, memory manager, device drivers, process manager etc.) that perform basic required functions. In many operating systems, only the kernel can access hardware directly.

b) Command Interpreter

The command interpreter (processor) is the part of the operating system that understands and executes commands that are entered interactively by a human being or from a program. In some operating systems the command interpreter is called shell. It provides an interface between users and the operating system (kernel). When a user logs in, a shell is started up.

c) SystemCalls

The interface between the OS and the user program is defined by a set of instructions called **system calls**. User programs use system calls to talk with the operating system.

***Definition**: System call is the mechanism by which a program requests a service from the operating system's kernel.*

When a running program needs a service from the operating system, it calls a system call.

4.5.2 Memory Management

For a program to be executed, it must be found in main memory (RAM). In a multiprogramming environment in which several programs can reside in memory at the same time, every program and its data must be protected from the actions of other programs. Memory management keeps track of what programs are in memory and where in memory they reside.

Memory allocation determines where a program as well as its related data will be kept in memory for execution. Memory needs to be allocated efficiently to pack as many jobs in memory as possible. Memory can be subdivided into segments or frames (pages).

a) Segmentation

Segmentation is when memory is divided into variable sized units called segments. When segments are used, memory allocation can be done in three different ways:

i. First fit allocates the first free segment that is large enough for the new process.
ii. Best fit allocates the smallest block among those that are large enough for the new process.

iii. Worst fit allocates the largest block among those that are large enough for the new process.

b) Paging

Paging is when memory is divided into fixed-size units called frames. Jobs are broken up into blocks of same size as frames called pages which are allocated a number of frames. The OS then uses a *page table* to map program pages to memory frames. The pages for each job could be in logical order or they may be scattered about wherever there is a free frame.

c) Virtual Memory

A program may require more memory than it is available. To solve this problem, virtual memory is used. Virtual memory is part of the hard disk that is used as an extension of RAM. It is slower, but it is considerably bigger. As execution goes on, data is being swapped between RAM and virtual memory.

When a program is running, only the pages that contain the necessary data are kept in RAM while those that are not needed are kept on disk. For example, a program that has been minimized for a long time may be transferred to virtual memory so as not to fill up the main memory.

Disk thrashing occurs when the OS has to spend a considerable proportion of its time swapping data between virtual and real memory.

4.5.2 Process Management

A process is a program in execution. It consists of the program's instructions and the resources allocated to it for execution. A program is static while a process is active. The operating system performs process management to carefully track the progress of a process and all of its intermediate states.

A process changes state as it executes. The different states a process can have are shown in the diagram below.

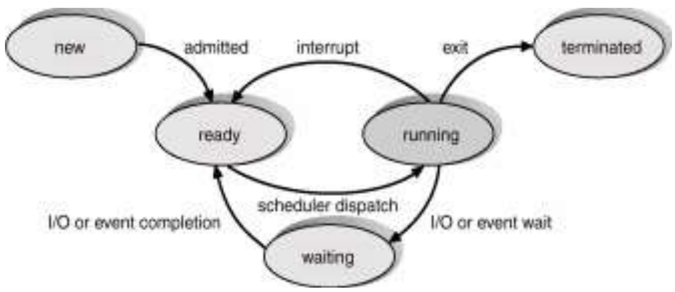


Figure 4.2: Process states

1. **New** - The process is being created.
2. **Ready** - Process has all needed resources - waiting for CPU only.
3. **Running** - Instructions are being executed.
4. **Waiting** - Process is waiting for some event to occur (human, hardware or another process)
5. **Terminated** - Process has finished execution

Definition*: The turnaround time for a process is the amount of time between the time the process arrives the ready state and the time it exits the running state for the last time.*

In other words, it is the length of time it takes to run a process from initialization to termination, including all the waiting time.

a) Process Control Block

The operating system manages a large amount of data about a process like the program counter (PC), CPU registers, memory management information, I/O status, scheduling data and process state. This data is stored in a data structure called a process control block (PCB) also called process descriptor or state vector. Every process has its PCB and each time a process is moved to the running state, its register values are loaded into the CPU while register values for the currently running process are stored into its PCB. This exchange of information is called a context switch.

b) Process Synchronization

In a multi-tasking system, processes compete for resources. A resource is anything that is required by a process to accomplish its task (processor, memory, I/O device, bus, file etc). Some resources can only be used in a non-sharable or exclusive mode. That is, they cannot be used by more than one process at a time. Such resources are known as critical resources. A critical section is a part of a program where it has access to a non-sharable (critical) resource. To prevent two or more processes from entering their critical sections over the same resource, processes must be synchronized.

***Definition**: Process synchronization is about getting processes to coordinate together in order to avoid two or more processes entering into critical section over the same resource.*

If processes are not synchronized, it leads to deadlock and starvation.

c) Deadlock

Deadlock is a permanent blocking of a set of processes competing for resources. It is a situation in which each of two processes is waiting for the other to do something; thus, neither one can proceed. For a deadlock to occur, the following four conditions must hold.

- ✓ **Mutual Exclusion**: At least one resource must be held in a non-sharable way.
- ✓ **Hold and Wait**: A process must be holding a resource and waiting for another.
- ✓ **No Pre-emption**: No resource can be forcibly removed from a process holding it.
- ✓ **Circular Wait:** A waits for B, B waits for C, C waits for A.

Deadlock can be prevented by ensuring that one of the above conditions does not hold. For example using pre-emptive scheduling.

d) Starvation

Starvation is a situation where a task can never finish because it can never get a necessary resource such as a large block of memory. The operating system should detect such tasks and do its best to allocate the resources that they need.

4.5.3 Process/CPU Scheduling

CPU scheduling is the act of determining which process in the ready state should be moved to the running state. It decides which process in memory is to be executed by the CPU at any given moment. Scheduling decisions may take place when a process:

1. Switches from running to waiting state
2. Switches from running to ready state
3. Switches from waiting to ready
4. Switches from running to terminated

Scheduling can be pre-emptive or non-pre-emptive.

1. **Pre-emptive** scheduling is scheduling in which the currently executing process is forced to give up the CPU. Scheduling under (2) and (3) is pre-emptive.

2. **Non pre-emptive scheduling** is scheduling in which the currently executing process gives up the CPU voluntarily. Scheduling under (1) and (4) is non-pre-emptive.

There exist different algorithms used for scheduling. Examples are first come, first served, shortest job first and round robin algorithms.

a. First Come, First Served

First come first served algorithm moves processes to the CPU in the order in which they arrive in the ready queue. It is non-pre-emptive. As such, when a process has the CPU it runs to completion before giving up the CPU.
Consider that the following processes arrive in the order they are given below.

Process	Service time
P1	
	75
	320
P4	280
	125

Table 4.1: showing processes to be executed

Using FCFS algorithm to schedule the processes, we get

Figure 4.3: FCFS algorithm implemented using processes in Table 4.1

b. Shortest Job First

Shortest job first (SJF) algorithm looks at all the processes in the ready state and dispatches the one with the smallest execution time. It is also generally implemented as a non-pre-emptive algorithm. Using the SJF algorithm, we have

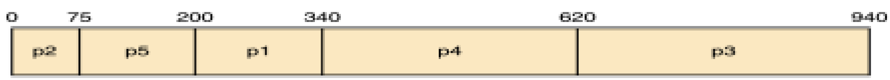


Figure 4.4: SJF algorithm implemented using processes of Table 4.1

Disadvantage

- Some jobs may never be executed as they have a high service time (starvation).

c. Round Robin

Round-robin algorithm distributes time equitably among all ready processes by establishing a particular time slice (or time quantum), during which each process executes. At the end of the quantum, the process is pre-empted. It returns to the ready state to allow another process its turn. Using round-robin algorithm with time slice of 50, the above processes will be scheduled as follows:

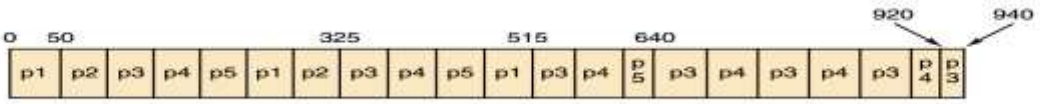


Figure 4.5: *Round Robin algorithm implemented using the processes of* ***Table 4.1***

4.5.4 Management of I/O Devices

a) Interrupts: An interrupt is a signal from a device attached to a computer or from a program within the computer that causes the main program that operates the computer (the operating system) to stop and figure out what to do next. Almost all personal (or larger) computers today are ***interrupt-driven*** - that is, they start down the list of computer instruction s in one program (perhaps an application such as a word processor) and keep running the instructions until either (A) they can't go any further or (B) an interrupt signal is sensed. After the interrupt signal is sensed, the computer either resumes running the program it was running or begins running another program.

Basically, a single computer can perform only one computer instruction at a time. But, because it can be interrupted, it can take turns in which programs or sets of instructions that it performs. This is known as multitasking. It allows the user to do a number of different things at the same time. The computer simply takes turns managing the programs that the user effectively starts. Of course, the computer operates at speeds that make it seem as though all of the user's tasks are being performed at the same time. (The computer's operating system is good at using little pauses in operations and user think time to work on other programs.)

An operating system usually has some code that is called an ***interrupt handler***. The interrupt handler prioritizes the interrupts and saves them in a queue if more than one is waiting to be handled. The operating system has another little program, sometimes called a scheduler, that figures out which program to give control to next.

In general, there are hardware interrupts and software interrupts. A hardware interrupt occurs, for example, when an I/O operation is completed such as reading some data into the computer from a tape drive. A software interrupt occurs when an application program terminates or requests certain services from the operating system. In a personal computer, a hardware interrupt request (IRQ) has a value associated with it that associates it with a particular device.

i) Interrupt Latency (Priorities): Interrupt latency, also called interrupt response time, is the length of time that it takes for a computer interrupt to be acted on after it has been generated. In most computers, a trade-off exists among interrupt latency, throughput, and processor utilization.

Factors that affect interrupt latency include the microprocessor design (or architecture), the microprocessor clock speed, the particular OS employed, and the type of interrupt controller used. Minimum interrupt latency depends mainly on the configuration of the interrupt controller, which combines interrupts onto processor lines, and assigns priority levels to the interrupts. Maximum interrupt latency depends mainly on the OS.

b) Polling: In electronic communication, 'polling' is the continuous checking of other programs or devices by one program or device to see what state they are in, usually to see whether they are still connected or want to communicate.

Specifically, in multipoint or multi-drop communication (a controlling device with multiple devices attached that share the same line), the controlling device sends a message to each device, one at a time, asking each whether it has anything to communicate (in other words, whether it wants to use the line).

c) Buffering: A buffer is a data area shared by hardware devices or program processes that operate at different speeds or with different sets of priorities. The buffer allows each device or process to operate without being held up by the other. In order for a buffer to be effective, the size of the buffer and the algorithms for moving data into and out of the buffer need to be considered by the buffer designer. Like a cache, a buffer is a "midpoint holding place" but exists not so much to accelerate the speed of an activity as to support the coordination of separate activities.

This term is used both in programming and in hardware. In programming, buffering sometimes implies the need to screen data from its final intended place so that it can be edited or otherwise processed before being moved to a regular file or database.

4.5.5 File Management

The file system is the portion of the operating system that manages how files are stored. Examples of file systems are FAT used in MS DOS, NTFS used in Windows, ext2 used in Linux and HPFS used in OS/2. The operating system is responsible for the following activities in connection with file management:

1. File creation and deletion.
2. Directory creation and deletion.
3. Support of primitives for manipulating files and directories
4. Mapping files onto secondary storage.
5. File backup on stable (non-volatile) storage media.

4.5.6 Providing User Interface

Working on a computer, a user has to interact with the computer. A user interface is the means of communication or interaction between the user and the computer. The operating system provides this means that enables an individual to see and work when using a computer.
Different operating systems provide different types of user interfaces.

a) Command Line Interface (CLI)
A command line interface allows the user to interact with the computer by typing the commands in a specified format. It provides a prompt through which the user types the commands. Here the user mostly makes use of the keyboard.
In this type of interface, the user has to remember the name and format of the commands. Spelling mistakes and deviations in format lead to errors and the task is not performed. Examples of operating systems that provide a command line interface are MS DOS, early versions of UNIX and Linux.

b) Graphical User Interface (GUI)
A graphical user interface allows the user to interact with the computer through graphical items such as icons, menus, dialog boxes, etc. Here, the user mostly makes use of the mouse to point and click on these graphical items. GUI is also known as WIMP system where WIMP stands for windows, icons, menus and pointers.
This type of user interface requires a lot of memory space to store the graphics and can cause machines with low processing power to be slow. Examples of operating systems with a graphical user interface are Windows, Macintosh operating systems, some versions of Linux and UNIX.

c) Voice Recognition Interface: A voice/speech recognition interface allows the user to give verbal commands to the computer. The user communicates with the computer through natural language. They are also called natural language interface

4.5.7 Providing Security

Security is the most desirable characteristic of any operating system. An operating system should provide a means for safeguarding system resources from unauthorized users and protection of one user's resources from other users of the system.
Popular operating systems offer security features through incorporation of the following:

a) Login name.
b) Login password.
c) Read/write access file permissions.
d) Encryption of data
e) Virus protection software layers.

4.5.8 Types of Operating Systems

a) SingleUser OS
A single user operating system is an operating system which allows only one user to work on the system at a time. No two or multiple users can work on the system simultaneously.

Examples are Control Program for Microcomputers (CP/M) and Microsoft Disk Operating System (MS DOS).

b) MultiUser OS

A multi-user operating system is an operating system which allows multiple users to work on the system simultaneously. This type of operating system is larger and more complex than a single user operating system.

Features of multi-user operating systems, which are not provided in single user operating systems, are:

- Time sharing (CPU devotes time to all the users in round robin fashion).
- Tight security features.
- Resource sharing among users.
- System administrator privileges

Examples of multi-user operating systems are Linux, Unix and Virtual Machine System (VMS).

c) Singletask OS

A single task operating system allows a user to execute one and only program at a time. The user cannot run two or more programs at the same time. It is not possible to be preparing a worksheet on the computer while printing a report or listening to music. Once a user invokes a program, the computer gets dedicated to that task only.

CP/M and MS-DOS are examples of single user single task operating systems.

d) Multiprogramming OS

A multi-programming operating system is an operating system that allows multiple programs to be held in main memory at the same time. The concept of multi-programming is that the operating system keeps several jobs in memory simultaneously and decides which can be executed at a given moment.

e) Multitasking OS

A multitasking operating system allows a user to execute more than one program at a time. It allows a user to be preparing a worksheet on the computer while printing a report or listening to music. Multitasking is an extension of multiprogramming as two programs cannot be executed simultaneously if they are not found in memory at the same time.

Windows Me, Windows-XP, Macintosh operating system, OS/2 are examples of single user, multitasking operating systems.

f) Embedded OS

An embedded operating system is an operating system that is used in an embedded system. An embedded system is a small computing device that is built into a larger equipment often as a single chip and dedicated to a given task. Embedded systems control many devices in use today such as digital watches, mobile phones, microwave ovens, washing machines, vehicles, photocopiers, cameras and process controllers.

Embedded OS are ROM based. That is, they cannot be modified as ROM is read only.

g) Network OS

A network operating system is an operating system which includes networking features. It contains special functions, protocols and device drivers that enable the computer to be connected to a network.

Examples of network operating systems are Windows-NT, Windows-2000 server, Windows server 3000, Novell Netware and Artisoft LANstatic.

Some multi-purpose operating systems like Windows XP, Windows 7 and Mac OS 10, come with capabilities that enable them to be described as network operating systems.

4.6 Application Software

Application software is computer software that causes a computer to perform useful tasks beyond the running of the computer itself. They provide user-oriented functionality. Application software are used to carry out productive work like typing a letter, designing and invitation card, surfing the Internet, listening to music or watching a movie. Application software includes a variety of programs that can be subdivided into general purpose software, special purpose software and custom-written software.

4.6.1 General Purpose Software

General purpose software is designed for a variety of tasks. It is not limited to one particular function. They are also called generic software and can be found off-the-shelf. That is, they can be bought from the market. Different types of general-purpose application software exist.

a. Word Processing Software

Word processing software is used to prepare and edit text. Word processors are used to create all types of documents such as letters, reports and essays. Using a word processor you can type, revise and correct a document on the screen before printing it out. The layout of the page can be changed and a wide variety of different styles of text can be used to improve the presentation of a document. Tools such as spelling checkers can help ensure that the content of a document is accurate.

Examples are MS Word, WordPerfect, WordPad, and Notepad

b. Spreadsheet software

A spreadsheet is a table of values arranged in rows and columns. Each value can have a predefined relationship to the other values. If you change one value, therefore, you may need to change other values as well. Therefore, *Spreadsheet applications* (sometimes referred to simply as *spreadsheets*) are computer programs that let you create and manipulate spreadsheets electronically. In a spreadsheet application (software), each value sits in a cell. You can define what type of data is in each cell and how different cells depend on one another. The relationships between cells are called *formulas*, and the names of the cells are called *labels*.

Examples are MS Excel, Lotus 1-2-3, Quattro Pro, Apple Numbers

c. Database Software

Database software is the phrase used to describe any software that is designed for creating databases and managing the information stored in them. Sometimes referred to as *database management systems* (DBMS), database software tools are primarily used for storing, modifying, extracting, and searching for information within a database. Database

software is used for a number of reasons in any industry - from keeping your bookkeeping on task, compiling client lists to running your online Web site. Because they have so many uses, there are dozens of database software programs available. Examples are MS-Access, Oracle, Dbase, FileMaker Pro.

d. Presentation Software

Presentation software is a category of application software that is specifically designed to allow users to create a presentation of ideas by stringing together text, images and audio/video. The presentation tells a story or supports speech or the presentation of information.

In the same token, Presentation software can be divided into business presentation software and general multimedia authoring software, but most presentation software applications already provide tools that allow users to create both professional-looking business presentations and general multimedia presentations. Examples are MS PowerPoint, Hyper Studio, Digital Chisel

e. Desktop Publishing Software

Desktop publishing is the process of using the computer and specific types of software to combine text and artwork to produce documents properly formatted for print, Web, or mobile devices such as newsletters, brochures, books,business cards, Web pages, greeting cards, letterhead, packaging, signage, etc. Examples are MS Publisher, Adobe PageMaker, Quark Express.

f. Integrated Software

An integrated software package is a software that consists of multiple applications bundled together. They usually have related functions, features and user interfaces, and may be able to interact with each other. Examples are Microsoft Office, Open Office, Lotus Smart suite and Microsoft works.

4.6.2 Special Purpose Software

Special purpose software is designed for a specific task. Examples are web browsers, game applications, weather forecasting applications, media players and flight control software.

4.6.3 Custom-Written Software

Custom-written software is written to a particular user's requirements or needs. It is generally not available off-the-shelf and often has to be purchased directly from the software manufacturer. It is also called bespoke software or tailor made software as it is tailored to the exact requirements of the user or organization.

4.7 Software Distribution Licenses

Software are distributed under different licenses. They include public domain, freeware, shareware, open source and all rights reserved.

a) Public Domain

Public domain software has no owner and is not protected by copyright law. It was either created with public funds or the ownership was forfeited by the creator. Public domain software can be copied, sold and/or modified. It is often of poor quality.

b) Shareware

Shareware is copyrighted software that allows you to use it for a trial period but you must pay a registration fee to the owner for permanent use. Purchasing the right to use the software may also get you a version with more powerful features and published documentation.

c) Freeware

Freeware is copyrighted software that is licensed to be copied and distributed without charge. Freeware is free but it is still under the owner's control. Examples are Eudora and Netscape.

d) **Commercial Software**

All rights reserved software is software that must be used by the purchaser according to the exact details spelt out in the license agreement.

e) Open Source Software

Open source software is software whose source code is published so that a variety of people can add contribution. Examples are Linux OS, MySQL, Mozilla and OpenOffice.

Drill Questions:

1. Define the terms scheduler and dispatcher.

Answer*:*

i) *Scheduler: the part of the kernel in charge with the strategy for allocation/de-allocation of the CPU to each competing process. It maintains a record of all processes (called process table) in the OS, adds new processes to the process table and removes the ones that are completed.*
ii) *Dispatcher: the component of the kernel that oversees the execution of the scheduled processes. It performs "process switch" – procedure to change from one process to another.*

2. (i) Briefly describe the following system software stating the category of each

(a) Compiler (b) Defragmenter (c) File compression (d) Interpreter

(ii) State four functions of an operating system when managing a computer's resources

(iii) Define the term interrupt. State one scenario in which an interrupt is used in the management of an operating system.

(iv) Name three important factors to consider when designing a new Computer User Interface. (**Q7, CGCE 2015**)

3. Name three factors to consider when designing a new computer user interface.

4.(i) Define the following terms: (a) Process (b) Interrupt (c) Deadlock

(ii) Define the term interrupt. State one scenario in which an interrupt is used in the management of an Operating system.

(iii) List two major activities of an operating system with respect to:

(a) Memory management (b) Secondary storage management

(iv) Differentiate between Multiprocessing and multiprogramming. (**Q7iii;Q9i, CGCE 2015**)

5.(i) Descirbe the main difference between applications software and systems software.

(ii) Name four types of system software and give one use for each class.
(iii) Name the type of application software you would use for the following purposes.
(a) To create, edit and format text based documents.
(b) To work with numbers, calculations, charts and graphs.
(c) To organise and access large amounts of information.
(d) To produce good news letters and company magazines. (**Q5, CGCE 2014**)

6.(i)(a) What is an Operating system?
(b) Esplain the role played by the operating system in the given instances
- Integrating additional hardware to a computer system
- Managing RAM usageduring processing.

(c)(a) What is a computer interface?
(b) Give the characteristic of the graphical User interface.
(c) State two advantages of GUI over Command Line. (**Q4i,ii, CGCE 2016**)

CHAPTER 5: FILE FORMAT AND ORGANISATION

5.1 Data Hierarchy

Data items processed by computers form a data hierarchy in which they become larger and more complex in structure as we progress from bits, to characters, to fields and to larger data elements. Data elements can be organized in a hierarchical form as follows:

Bit ➔ Byte (character)➔ Field ➔Record ➔ File ➔ Database

i. **Bit:** The smallest data item manipulated by computers is the bit (short for binary digit). Each bit can assume either the value 0 or the value 1.

ii. **Byte:** A single bit is of little or no use as it can represent only two states. To make them important, they are usually used in groups. A group of eight bits is known as a byte. A byte is the smallest addressable unit of the computer and it is used to represent a character. Each character, be it a letter, a digit or a symbol is represented as an eight bits pattern. The word "data" has 4 characters, meaning that it will take up 4 bytes which is 32 bits.

iii. **Field:** As characters are composed of bits, fields are composed of characters. A field is a group of characters that conveys some meaning. For example, a person's name, a word, the name of a place or thing.

iv. **Record:** A record is a collection of related fields. A record consists of fields, with each field describing an attribute of an entity.

v. **File:** A file is a collection of related records. Files are frequently classified by the application for which they are primarily used. A primary key in a file is a field whose value identifies a record among others in the file.

vi. **Databases:** A database is a collection of files.

5.2 File Formats

Records in a file must be stored in a way that a program that uses the file will be able to recognize and possibly access it. This is known as file format.

***Definition**: File format is a particular way that data is encoded for storage in a computer file.*

A particular file format is often indicated as part of a file's name by a file name extension. Conventionally, the extension is separated from the file name by a dot and contains three or four letters that indicate the format. A file name is a unique name that is given to a file within its file directory. For example, lectures.docx is a file with filename "lectures" and extension "**.docx**" indicating that the file is an MS Word 2007 document.
There are as many different file formats as there are different programs to process the files. A few of the more common file format types are:

5.2.1 Graphics File Formats

Computers store graphic images as either bitmap images or vector graphics.

i) Bitmap Images

A bitmap image is stored as a collection of tiny dots (pixels) of individual colours that make up the image. Pixel is short for picture element. A data file for a bitmap image contains information about every single pixel in the image. As a result, the file size of a bitmap graphic is often quite large.

Bitmap images are resolution dependent. Resolution refers to the number of pixels in an image and is usually stated as **dpi** (dots per inch) or **ppi** (pixels per inch). The more the number of pixels in an image, the more detailed the image will be.

Because bitmaps are resolution dependent, it's difficult to increase or decrease their size without sacrificing a degree of image quality.

Some common bitmap graphic programs are:

i. Photoshop
ii. Paint Shop Pro
iii. GIMP
iv. Photo-Paint
v. Graphic Converter

Examples of bitmap formats include:

a) **JPEG** or **JPG** - Joint Photographic Experts Group
b) **GIF** - Graphics Interchange Format
c) **TIFF** or **TIF**- Tagged Image File Format
d) **BMP** - Bitmap graphics file
e) **PNG** – Portable Network Graphic

ii) Vector Graphics

Vector images are stored as a collection of shapes (lines, circles, curves) called objects, together with information defining how the objects will be produced and where they will be located. The data file for a vector image contains the points where the shapes start and end, how much the shapes curve, and the colors that either border or fill the shapes. Because vector graphics are not made of pixels, the images can be scaled to be very large without losing quality. Programs used with vector graphics are drawing programs. Some of these programs include:

1. Corel Draw
2. Adobe Illustrator
3. Adobe Photoshop

Examples of vector formats are:

i. **SVG** - Scalable Vector Graphic
ii. **CDR**- CorelDraw graphic
iii. **CMX** - Corel Meta Exchange
iv. **EPS** - Encapsulated Postscript
v. **CGM** - Computer Graphics Metafile

vi. **PICT** - Macintosh Picture
vii. **WMF**- Windows Metafile

5.2.2 Multimedia File Formats

Multimedia is any combination of digitally manipulated text, sound, animation and video.

a) Audio File Formats

i. **MP3**- MPEG Layer 3
ii. **WMA**- Windows Media Audio
iii. **WAV** - Waveform audio file

b) Video File Formats

i. **AVI** - Audio Video Interleave
ii. **MPEG** or **MPG** - Motion Picture Experts Group
iii. **3GPP** - Third Generation Partnership Project

5.2.3 Common Application File Formats

i. **DOC** or **DOCX** - Document file (ASCII or MS Word)
ii. **PDF** - Portable Document Format
iii. **TXT** - ASCII Text file
iv. **XLS (XLSX**)- Excel Worksheet file
v. **WKS, WK2, WK3**- Lotus 1-2-3 or MS Works Worksheet
vi. **PPT (PPTX)-** PowerPoint presentation file

a) Hypermedia File Formats

i. **HTML (HTM)-** Hypertext Mark-up Language
ii. **XML**- Extensible Mark-up Language
iii. **SGML**- Standard Generalized Mark-up Language

b) Other File Formats

i. **EXE** - Executable file (machine-language program)
ii. **DLL** - Dynamic Link Library
iii. **SYS** - System file
iv. **ZIP**- Zip compressed file

5.3 File Organization

File organization refers to the logical structuring of records in a file. In other words, it is the way records are arranged in a file.
In the context of data processing, files can be grouped into two types: master and transaction files.
1.) A master file contains the permanent data of a data processing system. Master files represent the most up to date situation of the system and contain two basic types of data:

a) Data of a more or less permanent nature which only requires updating occasionally;

b) Data which will change every time transactions are applied to a file.

2. A transaction file contains data that is necessary to keep a master file up to date. Data is collected on a daily, weekly or monthly basis into a transaction file which is then used to update the master file.

Depending on the arrangement of records in a file, files can be serially, sequentially or randomly organized.

5.3.1 Serial File Organization

Serial files are files in which records are stored in chronological order with no particular sequence. As each record is received it is stored in the next available storage space. In order to access a particular record, the file must be read record by record from the beginning of the file until the required record is found. This type of access is called serial access. Serial access is very slow which makes it impossible for serial files to be very useful if individual records have to be looked up. They are used as temporary files to store transaction data.

5.3.2 Sequential File Organization

A sequential file is a serial file in which records are stored in order by a record-key field. That is, the records are sorted according to a given field that identifies each record in the file, in a unique way. Sequential files are also accessed serially. However, a sequential file is particularly useful when a whole file of data has to be processed as it is faster and more efficient than other methods. Sequential files are ideal for master files and batch processing applications such as payroll and billing systems in which almost all records are processed in a single run of the application.

Serial and sequential files can be stored on both magnetic disk and magnetic tape. They are particularly suited to tape which is a serial/sequential access medium.

5.3.3 Indexed Sequential File Organization

An indexed sequential file is a sequential file in which an index is added to support random access. In this type of file, records are stored in blocks and the address of every block is stored in an index. A block address may consist of the key of the first record in the block. Indexed sequential files therefore combine the advantages of a sequential file with the possibility of direct access. To access any record in the file, the index is searched to determine the block to which the record belongs. The record is then searched for sequentially but only within the block in which it is found.

5.3.4 Random File Organization

A random file is a file in which records are stored in random order. In a random file, the value of the record key is mapped by a hash function to an address within the file where the record will be stored. The function transforms the value of a record key into an index that corresponds to a location in the file. This process is called hashing. Any record within this type of file can be accessed directly without reference to any other record.

To create and maintain a random file, a mapping function must be established between the record key and the address where the record is held

5.4 Processing Files

The main processes that can be performed on files are adding records, deleting records and updating records.

5.4.1 Processing Serial Files

a) Adding a Record

As there is no particular order in a serial file new records are simply added at the end of the file. The process can be expressed in simple algorithm:

```
Open file A for writing
Move record pointer to end of file A
Write new record to file A
Close file A
```

b) Deleting a Record

The process of deleting a record from a serial file is more complex. It involves reading the records from the beginning of the file and writing them, one by one, to a new file and simply not writing the record chosen for deletion.

```
Open file A for reading
Open file B for writing
While not at end of file A
 Read a record from file A
 If Key_of_A<>Key_to_Delete then
  Write record A to File B
 Endif
Endwhile
Close file A
Close file B
```

File B now contains the amended data and file A still exists with the original data.

5.4.2 Processing Sequential Files

a) Adding a Record

In order to maintain the order of the sequential file the new record has to be placed in the correct position in the file. This is done by copying the records from the original file into a new file until the position for the new record is found. The new record is written and then the remaining records copied after.

```
Open file A for reading
Open file B for writing
While not at end of file A
 Read a record from file A
 If Key_of_A<=Key_to_Add then
  Write record A to File B
 Else
  Write new record to File B
```

```
  Endif
Endwhile
Close file A
Close file B
```

b) Deleting a Record

This is exactly the same process as for a serial file.

5.5 File Data Security

There is always a chance that data can be compromised though it may appear secure when confined in a computer. One could suddenly be hit with a malware infection where a virus destroys the files or hackers may gain access into the computer system and steal, delete or destroy files. To secure data, some measures need to be taken.

Data security is the practice of keeping data protected from corruption and unauthorized access. Measures taken to ensure the safekeeping of data include backup storage, archival storage, data encryption, authentication, logs, antivirus and firewall.

i) Backup Storage

Backup is the process of making copies of data which may be used to restore the original data after a data loss event. Copies of data can be made and stored on storage devices like external hard drives, CDs and DVDs. Backups have two distinct purposes; to recover data after its loss, be it by data deletion or corruption and to recover data from an earlier time, according to a user-defined data retention policy, typically configured within a backup application.

ii) Archival Storage

Data archiving is the process of moving data that is no longer actively used to a separate data storage device for long-term retention. Data archive is often confused with data backup, which are copies of data. Data archives protect older information that is not needed for everyday operations but may occasionally need to be accessed while data backups are used to restore frequently used data in case it is corrupt or destroyed.

5.6 Audit Logs

An audit log is a record of all the actions carried out by the user of a system, stored together with their user name. If a file is deleted, modified or copied, the person responsible can be identified by examining the logs. Getting a computer system to keep logs is one way of deterring authorized users from damaging or stealing data from the system.

i) Transaction Logs

A transaction log (also database log or binary log) is a history of actions executed by a database management system to guarantee ACID properties over crashes or hardware failures. ACID is a set of properties that guarantee that database transactions are processed reliably. ACID stands for atomicity, consistency, isolation and durability.

5.7 Atomicity

Atomicity requires that database modifications must follow an “all or nothing” rule. This means that, if one part of a transaction fails, the entire transaction fails and the database state is left unchanged. This guarantees that a transaction cannot be left in an incomplete state.

i) Consistency

Consistency ensures that any transaction will bring the database from one valid state to another. Any data written to the database must be valid according to all defined rules.

ii) Isolation

Isolation refers to the requirement that no transaction should be able to interfere with another transaction. No transactions that affect the same rows can run concurrently since their sequence and hence their outcome would be unpredictable.

iii) Durability

Durability means that once a transaction has been committed, it will remain so even in the event of power loss, crashes or errors.

5.7 Virus Checking Software

A virus checking software or antivirus is software that is used to prevent, detect and remove malware. Malware are malicious software including but not limited to computer viruses, computer worms, Trojan horses, spyware and adware. Malware are designed to cause damage or disruption to a computer system. When an antivirus has been installed on a system, it is necessary to update it regularly. Examples of antivirus software are Avast, Kaspersky, McAfee, Avira, AVG and Eset NOD32.

Drill Questions:

1.(a) There are two different kinds of graphics stored in a computer: bit-mapped graphics and vector based graphics. Explain the meaning of bit-mapped graphics and vector-based graphics, giving one file format (extension) used for each.

(b) Give three advantages of vector-based graphics over bit-mapped graphics. (**Q3i, CGCE 2014**)

2.(a) What is a file format? Give one type of file format.

(b) Explain one importance of file formats.

(c) State two reasons why file formats vary.’ (**Q4i, CGCE 2015**)

Chapter 6: NUMBER SYSTEMS AND CODE

A number system is a set of symbols and rules used to represent numbers. The number of different symbols used in a given number system is known as the base or radix of the system. The largest value of a symbol (digit) in a given number system is always one less than the base or radix of that system. If the base of a system is represented by"b", then the largest value a digit in that system can assume is "$b - 1$".
Examples of number systems include the binary, octal, decimal (denary) and hexadecimal systems.

Most Significant Bit (MSB)	2^{n-1}	…	2^2	2^1	2^0	Least Significant Bit (LSB)

Figure 6.1: *Binary positions for an n-bit number*

6.1 The Decimal System

The decimal system has a base value of 10. Its maximum or largest value of digit is$(b - 1) = 10 - 1 = 9$, meaning thatthe decimal system uses the digits $0, 1, 2, 3, 4, 5, 6, 7, 8,$ and 9 to represent any quantity.
For any *n*-digit number, the value each digit represents depends on its position in the number. Decimal positions (place values) are powers of ten as shown below:

Most significant Digit (MSD)	10^{n-}	...p	10^2	10^1	10^0	Least Significant Digit (LSD)

Figure 6.2: *Decimal positions*

Where,10^0 is the ones place
$10^1$is the ten's place
$10^2$is the hundreds place

The value of any digit in a given *n*-digit number is obtained by multiplying the digit by its place value. The value of the number is the sum of the products of the digits and their place values. That is, we multiply each digit by its place value, then we add the different products obtained.
Example: $\mathbf{234_{10} = 2 \times 10^2 + 3 \times 10^1 + 4 \times 10^0}$ This means that there are two one hundreds, three tens and four ones in the number 234.

Remark Each digit position in decimal has a weight that is ten times the one to its immediate right. That is, $10^2$is ten times greater than 10^1 which is ten times greater than 10^0.

6.2 The Binary System

The binary system has a base value of 2. Only two digits 0 and 1, are used to represent any quantity in binary. These digits are called binary digits or more commonly bits. To express any quantity in binary we use powers much like in the decimal system but this time, the weights are powers of 2.

Example: the binary number 1101_2 is expressed as $1101_2 = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$

This means that there are 1 one, 0 two, 1 four, and 1 eight in the number 1101.

6.2.1 Binary Addition

Rules for addition:

- 0 + 0 = 0
- 0 + 1 = 1
- 1 + 0 = 1
- 1 + 1 = 0, and carry 1 to the next more significant bit

Example (i) 00011010 + 00001100 = 00100110

1 1 carries
0 0 0 1 1 0 1 0 = 26(base 10)
+ 0 0 0 0 1 1 0 0 = 12(base 10)
0 0 1 0 0 1 1 0 = 38(base 10)

(ii) 00010011 + 00111110 = 01010001

1 1 1 1 1 carries
0 0 0 1 0 0 1 1 = 19(base 10)
+ 0 0 1 1 1 1 1 0 = 62(base 10)
0 1 0 1 0 0 0 1 = 81(base 10)

6.2.2 Binary Subtraction

Rules for subtraction:

- 0 - 0 = 0
- 0 - 1 = 1, and borrow 1 from the next more significant bit
- 1 - 0 = 1
- 1 - 1 = 0

Example(i): 00100101 - 00010001 = 00010100

0 borrows
0 0 $\cancel{1}^{1}$0 01 0 1 = 37(base 10)
- 0 0 01 0 0 0 1 = 17(base 10)
0 0 01 0 1 0 0 = 20(base 10)

(ii) 00110011 - 00010110 = 00011101

0^{1}0 1 borrows
0 0 $\cancel{110}^{1}$0 1 1 = 51(base 10)
- 0 0 010 1 1 0 = 22(base 10)
0 0 011 1 0 1 = 29(base 10)

6.2.3 Binary Multiplication

Rules for multiplication:

- 0 x 0 = 0
- 0 x 1 = 0
- 1 x 0 = 0
- 1 x 1 = 1, and no carry or borrow bits

Example:

(i) 00101001 × 00000110 = 11110110

```
          0 0 1 01 0 0 1   =   41(base 10)
        ×0 0 0 0 0 1 1 0   =   6(base 10)
        ----------------
         0 0 0 0 0 0 0 0
         0 0 1 0 1 0 0 1
       0 0 1 0 1 0 0 1
       -----------------
     0 0 0 1 1 1 1 0 1 1   =   246(base 10)
```

(ii) 00010111 × 00000011 = 01000101

```
          0 0 0 1 0 1 1 1   =   23(base 10)
       ×  0 0 0 0 0 0 1 1   =   3(base 10)
          -----------------
              1 1 1 1 1         carries
            0 0 0 1 0 1 1 1
          0 0 0 1 0 1 1 1
          -----------------
          0 0 1 0 0 0 1 0 1   =   69(base 10)
```

***Another Method**: Binary multiplication is the same as repeated binary addition; add The multicand to itself, the multiplier number of times.*

For example,

00001000 × 00000011 = 00011000

```
              1      carries
       0 0 0 0 1 0 0 0   =   8(base 10)
       0 0 0 0 1 0 0 0   =   8(base 10)
     + 0 0 0 0 1 0 0 0   =   8(base 10)
     ---------------------
       0 0 0 1 1 0 0 0   =   24(base 10)
```

6.2.4 Binary Division

Rules for division:

0/1 = 0; 1/1 = 1; 1/0 = undefined.

Binary division is the repeated process of subtraction, just as in decimal division.

Example:

00101010 ÷ 00000110 = 00000111

```
                        1 1 1     =   7 (base 10)
            -------------------
  1  1  0) 0  0  1¹0  1  0 1 0    =   42(base 10)
                - 1 1 0           =   6 (base 10)
               ---------------
                 1        borrows
                10¹0 1
                -1 1 0
                ----------
                     1 1 0
                   - 1 1 0
                ------------
                         0
```

Exercise: work this out 10000111 ÷ 00000101 (*apply the same method above and your anser should not be different from **00011011***)

6.3 The Octal System

The octal system has a base value of 8. It uses the digits 0-7 to represent any quantity. Weights in octal are powers of eight as shown below.

Most Significant Digit (MSD) →	8^{n-1}	…	8^2	8^1	8^0	← Least Significant Digit (LSD)

Figure 6.3: *Octal positions*

Where 8^0 is the one's place

$8^1$is the eights place

$8^2$is the sixty fours place

Example: $367_8 = 3 \times 8^2 + 6 \times 8^1 + 7 \times 8^0$

(This means that there are 3 sixty fours, 6 eight's and 7 one's in 367)

6.4 The Hexadecimal System

The prefix "hex" means 6 and "deci" means 10. The hexadecimal number system is thus a base-16 number system. Each digit position represents a power of 16. The digits used in this system are 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E and F, where$A = 10_{10},\ B = 11_{10},\ C = 12_{10},\ D = 13_{10},\ E = 14_{10},\ F = 15_{10}$.

6.5 Conversion from One Base to Another

6.5.1 Binary/Decimal Conversion

a) Decimal to Binary

To convert a binary number to decimal, we proceed as follows:

Step 1: starting with the 1s place, write the binary place value over each digit in the binary number to be converted.

Step 2: add up all the place values that have a "1" in them

Example 1: Convert 11010_2 to base 10

$$\begin{array}{ccccc} 16 & 8 & 4 & 2 & 1 \\ 1 & 1 & 0 & 1 & 0 \end{array}$$

$$16 + 8 + 2 = 26_{10}$$

Exercise 6.1: Convert: 1111_2 (*15*), 10011_2 (*19*), 10101_2(*21*) to base 10.

Example 2: Convert 1101.011_2 to base 10

$$\begin{aligned} 1101.011_2 &= 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} + 1 \times 2^{-3} \\ &= 8 + 4 + 0 + 1 + 0 + {}^1/_4 + {}^1/_8 \\ &= 13 + 0.25 + 0.125 \end{aligned}$$

$= 13 + 0.25 + 0.125$
$= 13.375$

Exercise 6.2: Convert the following binary numbers to decimal

iii) 1010.101_2
iv) 10011.001_2

b) Decimal to Binary

There are two methods that can be used to convert decimal numbers to binary:

- repeated division method
- repeated subtraction method

i) Repeated Division Method

The general technique of this method can be used to convert any decimal number to any other number system.

Step 1: divide the decimal number you want to convert by 2 in regular long division until you obtain a final remainder

Step 2: use the remainder as the least significant bit of the binary number

Step 3: divide the quotient you got from the first division operation by 2 until you obtain a final remainder

Step 4: use the remainder as the next digit of the binary number

Step 5: repeat steps 3 and 4 as many times as necessary until you get a quotient that cannot be divided by 2.

Step 6: use the last remainder (that cannot be divided by 2) as the most significant bit of the binary number.

Example 1: Convert $213_{10}$to base 2

$213/2 = 106$	remainder 1	LSB
$106/2 = 53$	remainder 0	↑
$53/2 = 26$	remainder 1	
$26/2 = 13$	remainder 0	
$13/2 = 6$	remainder 1	
$6/2 = 3$	remainder 0	
$3/2 = 1$	remainder 1	
$1/2 = 0$	remainder 1	MSB

Therefore, $213_{10} = 11010101_2$

***ii)* Repeated Subtraction Method**

Step 1: starting with the 1s place, write down all the binary place values in order until you get to the first value that is greater than the number to be converted.

Step 2: mark out the largest place value

Step 3: subtract the new largest place value from the number to be converted. Place a “1” under the place value.

Step 4: for the rest of the place values, try to subtract each one from the previous result. If you can, place a “1” under that place value. If you cannot, place a “0”.

Step 5: repeat step 4 until all the place values have been processed.
The resulting set of 1s and 0s is the decimal number you started with.

Example 2: 341_{10} to base 2

$512 \quad 256 \quad 128 \quad 64 \quad 32 \quad 16 \quad 8 \quad 4 \quad 2 \quad 1$
512 is greater 341 so we mark it out. What is left is
$256 \quad 128 \quad 64 \quad 32 \quad 16 \quad 8 \quad 4 \quad 2 \quad 1$
$341 - 256 = 85$ we place 1 under 256 (MSB)
$85 - 128 = x$ we place 0 under 128
$85 - 64 = 21$ we place 1 under 64
$21 - 32 = x$ we place 0 under 32
$21 - 16 = 5$ we place 1 under 16
$5 - 8 = x$ we place 0 under 8
$5 - 4 = 1$ we place 1 under 4
$1 - 2 = x$ we place 0 under 2
$1 - 1 = 0$ we place 1 under 1 (LSB)

Therefore, $341_{10} = 101010101_2$

Example 3: Convert 25.75_{10} to binary

$25_{10} = 11001_2$,
$0.75 \times 2 = 1.5 \quad 1$
$0.5 \times 2 = 1.0 \quad 1$
$0.75_{10} = 11_2$

$\Rightarrow 25.75_{10} = 11001.11_2$

Exercise 6.3: Convert: 33.33_2 to decimal.

6.5.2 Binary/Octal Conversion

Theorem

If base R_1 is the integer power of another base, R_2 (i.e. $R_1 = R_2^d$), then every group of d digits in R_2 is equivalent to 1 digit in the R_1 base.
For example:

Assume that: $R_1 = 8$ (octal) and $R_2 = 2$ (binary)

From the theorem, $8 = 2^3$
Hence, 3 digits in base-2 is equivalent to 1 digit in base-8.

From the stated theorem, the following is a binary-octal conversion table.

Binary	Hexadecimal
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

Table 6.1: *Binary to Octal conversion table*

a) Binary to Octal

***Step1*:** make groups of three bits starting from the least significant bit and move towards the most significant bit.

Step 2: replace each group of bits by its octal representation.

Example 1: Convert $100110_2$to base 8

$100110_2 = 100 \;\; 110$

$100_2 = 4_8, 110_2 = 6_8$

$\therefore 100110_2 = 46_8$

Example 2: Convert 1011101_2 to base 8

$1011101_2 = 001 \;\; 011 \;\; 101$

$001_2 = 1_8, \quad 011_2 = 3_8, \quad 101_2 = 5_8$

$\therefore \; 1011101_2 = 135_8$

b) Octal to Binary

To convert from octal to binary, we replace every octal digit by its 3-bits binary equivalent.

Example1: Convert 73_8 to binary

$7_8 = 111_2, \;\; 3_8 = 101_2$

$\therefore \; 73_8 = 111101_2$

Example 2: Convert 450_8 to binary

$4_8 = 100_2, \;\; 5_8 = 101_2, \;\; 0_8 = 000_2$

$\therefore 450_8 = 100101000_2$

6.5.3 Binary/Hexadecimal Conversion

Assume that:

$R_1 = 16$ (hexadecimal)

$R_2 = 2$ (binary)

From the theorem, $16 = 2^4$

Hence, 4 digits in a binary number is equivalent to 1digit in the hexadecimal number system.

The following is the binary-hexadecimal conversion table

Binary	Hexadecimal
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7

Binary	Hexadecimal
1000	8
1001	9
1010	A
1011	B
1100	C
1101	D
1110	E
1111	F

Table 6.2: *Binary to hexa-decimal conversion table*

a) Binary to Hexadecimal

Step1: make groups of four bits starting from the least significant bit and move towards the most significant bit.

Step 2**:** replace each group of bits by its hexadecimal value representation.

Example 1: Convert 100110_2 to base 16

$100110_2 = 0010 \quad 0110$

$0010_2 = 2_{16}, 0110_2 = 6_{16}$

$\therefore\ 100110_2 = 26_{16}$

Example 2: Convert 1011101_2 to base 16

$1011101_2 = 0000 \quad 0101 \ 1101$

$0000_2 = 0_{16}, \quad 0101_2 = 5_{16}, \quad 1101_2 = C_{16}$

$\therefore 1011101_2 = 5C_{16}$

b) Hexadecimal to Binary

To convert from hexadecimal to binary, we carry out the inverse operation. That is, we replace every hexadecimal digit by its 4-bits binary equivalent.

Example 1: Convert 450_{16} to binary

$4_{16} = 0100_2, \ 5_{16} = 0101_2, \ 0_{16} = 0000_2$

$\therefore 450_{16} = 10001010000_2$

Example 2: Convert $6E$ to base 2

$6 \ = \ 0110, \qquad E \ = \ 1110$

$\therefore\ 6E \ = \ 1101110_2$

Drill Questions

1) Convert the following octal numbers to hexadecimal.
   i) 65_8
   ii) 53_8

2) Convert the following hexadecimal numbers to octal.
   i) $12B_{16}$
   ii) $F2E_{16}$

6.6 Data Representation

Computers use binary patterns (fixed number of bits) to represent data, which could be numbers, letters, videos, images or other symbols. It is important to decide on how these patterns will be interpreted. The interpretation of binary patterns is called data representation or encoding. Different representation schemes exist.

6.6.1 Representation of Unsigned Integers

Unsigned integers can represent zero and all positive integers. The value of an unsigned integer is interpreted as "the magnitude of its underlying binary pattern". That is, for an unsigned integer, all the bits in the binary pattern are used to represent the magnitude of the integer.
In binary, an n-bit pattern can represent 2^{n} distinct integers. Therefore, an n-bit pattern will represent integers from 0 to $2^{\mathrm{n}} - 1$.

For example:
- A 4-bit pattern will represent the integers 0 to $(2^4) - 1 = 3$
- An 8-bit pattern will represent the integers 0 to $(2^8) - 1 = 255$

6.6.2 Representation of Signed Integers

Signed integers can represent zero, positive integers, as well as negative integers. Four representation schemes are available for signed integers:

1. Sign-Magnitude representation
2. 1's Complement representation
3. 2's Complement representation
4. Biased representation (Excess-n)

6.6.2.1 Sign-Magnitude Representation

In sign-magnitude representation, the most significant bit is the sign bit with 0 for positive and 1 for negative. The remaining $(n - 1)$ bits represent the absolute value of the integer. The absolute value of the integer is interpreted as "the magnitude of the $(n - 1)$-bit pattern".
Example 1: Suppose $n = 8$ and the binary representation 01000001.
Sign bit is 0 ⇒number is positive
Absolute value is $1000001 = 65$
Hence, the integer is 65.

Example 2: Suppose $n = 8$ and the binary representation 10001001.
Sign bit is 1 ⇒number is negative
Absolute value is $0001001 = 9$
Hence, the integer is -9

The drawbacks of sign-magnitude representation are:

- There are two representations (0000 0000 and 1000 0000) for the number zero, which could lead to inefficiency and confusion.
- Arithmetic is cumbersome making the design of electronic circuits for this scheme difficult.

6.6.2.2 1's Complement Representation

In 1's complement representation the most significant bit is still the sign bit. The remaining $(n-1)$ bits represent the magnitude of the integer as follows:

1. For positive integers, the absolute value of the integer is equal to "the magnitude of the $(n-1)$-bit pattern".
2. For negative integers, the absolute value of the integer is equal to "the magnitude of the complement (inverse) of the $(n-1)$-bit pattern".

Example 1: Using 1's complement, represent the following base 10 numbers on 8 bits.

i) $26_{10} = 11010$

Number is positive ⇒ sign bit is 0

⇒$26_{10} = 0\ 0011010$

ii) -25

Number is negative ⇒ sign bit is 1

$25 = 11001$

Complement of 0011001 is 1100110

⇒$-25 = 1\ 1100110$

Example 2: Give the decimal equivalent of the following 1's complement binary representations.

i) $0010\ 0001$.

Sign bit is 0 ⇒ positive

Absolute value is $010\ 0001 = 33$

Hence, the integer is $+33$

ii) $1000\ 0001$

Sign bit is 1 ⇒ negative

Absolute value is the complement of $000\ 0001$, i.e., $111\ 1110 = 126$

Hence, the integer is -126

The problem with this representation is that there are still two representations for zero (0000 0000 and 1111 1111).

a) Addition in 1's Complement

i. Add binary representations of the two numbers
ii. If there is a carry (referred to as end-round carry), add it to the result.

b) Subtraction in 1's complement

Subtraction is implemented using addition as follows:

i. Determine the 1's complement of the negative number
ii. Add the binary representations of the two numbers
iii. If there is any carry, add it to the result

Example 1: subtract 37 from 51 in binary

$37 = 0100101 \Rightarrow 37 = 1011010 \Rightarrow -37 = 1101101$in 1's complement

```
 (51)    00110011
(-37) + 11011010
        --------------
         1 00001101
       +          1
        ---------------
(14)     00001110
```

6.6.2.3 2's Complement Representation

In 2's complement representation, the remaining $(n-1)$ bits represent the magnitude of the integer as follows:

I. For positive integers, the absolute value of the integer is equal to "the magnitude of the $(n-1)$-bit pattern".
II. For negative integers, the absolute value of the integer is equal to "the magnitude of the complement of the $(n-1)$-bit pattern plus one". That is, we just add one to 1's complement to get 2's complement.

An *alternative* and simple way of getting 2's complement is

- to write the representation of the positive number
- Starting from the least significant bit, flip all the bits to the left of the first 1.

Example 1: Using 2's complement, store the following base 10 numbers.

i) -24

Number is negative ⇒ sign bit is 1

$24 = 11000 = 0011000$

1's complement of 24 is 1100111

2's complement of 24 is $1100111 + 1 = 1101000$

$\Rightarrow -24$ is represented by $1\ 1101000$

ii) -13

Number is negative ⇒ sign bit is 1

$13 = 1101 = 0001101$

1's complement of 13 is 1110010

2's complement of 13 is $1110010 + 1 = 1110011$

$\Rightarrow -13$is represented by $1\ 1110011$

Example 2: Give the decimal equivalent of the following 2's complement binary representations.

i) 01100000

Sign bit is 0⇒ positive

Absolute value is $110\ 0000 = 96$
Hence, the integer is $+96$

ii) $1\ 001\ 0001$
Sign bit is 1 ⇒ negative
Absolute value is the complement of $001\ 0001$ plus 1, i.e.,$110\ 1110 + 1 = 111$
Hence, the integer is -111 $(-2^7 + 2^4 + 2^0 = -128 + 16 + 1 = -111)$

Modern computers use 2's complement in representing signed integers. This is because:

1. There is only one representation for the number zero unlike in sign-magnitude and 1's complement representations.
2. Positive and negative integers can be treated together in addition and subtraction. Subtraction can be carried out using the "addition logic".

a. Addition in 2’s Complement

- Add binary representations of the two numbers
- If there is a carry, ignore it

b. Subtraction in 2’s Complement

Subtraction is implemented as follows:

- Determine the 2’s complement for the negative number
- Add the binary representations of the two numbers
- If there is any carry, ignore it

Example: subtract 37 from 51

$37 = 0100101 \Rightarrow 37 = 1011010$ (1’s C) $\Rightarrow -37 = 1\ 1011011$ (2’s C)

(51) 00110011
(-37) + 11011011
- ------------
(14) 1 00001110 discard end-round carry

6.6.2.4 Bias/Excess-k Representation

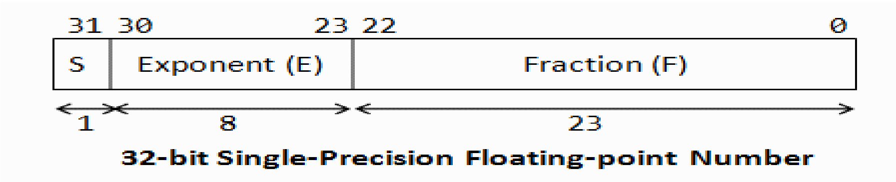
In excess-k notation, where n is the number of bits used, the value represented is the unsigned value with a fixed value 2^{n-1} subtracted from it.
In fact, excess K representation maps 0^n to -k, and 1^n to -k + (2^n – 1).

6.6.3 Representation of Real Numbers

Computers represent real numbers in a form similar to that of scientific notation of *mantissa* (M) and an *exponent* (E) of a certain *radix* (R), in the form of $M \times R^E$. Both M and E can be positive as well as negative. For example, 1.25×10^{-1} is expressed in scientific notation where 1.25 represents the mantissa, -1 the exponent and 10, the radix.
The IEEE 754 standard for representing floating-point numbers in the computer has two representation schemes: 32-bit single-precision and 64-bit double-precision.

✓ In 32-bit single-precision floating-point representation:

- the most significant bit is the *sign bit* (S), with 0 for negative numbers and 1 for positive numbers.
- the following 8 bits represent the *exponent* (E).
- the remaining 23 bits represent the *mantissa* (M) or *fraction* (F).



- In 64-bit single-precision floating-point representation:
  - the most significant bit is the *sign bit* (S), with 0 for negative numbers and 1 for positive numbers.
  - the following 16 bits represent the *exponent* (E).
  - the remaining 47 bits represent the *mantissa* (M) or *fraction* (F).

In floating-point representation, the mantissa and exponent play two important roles. The mantissa affects the precision of the number while the exponent affects the range. Precision has to do with the exactness of a number while range has to do with the set of numbers that can be represented.

For simplicity, we will use a 14-bit pattern for floating point with a 5-bit exponent and an 8-bit mantissa. In this case, we will use excess-16 representation for the exponent.

Examples:

1. Represent the following numbers as 32-bit single precision floating-point numbers.

a. 32.2 b. -15.37 c. 0.125

Solution:

a. ***Step 1**:* We convert the number to binary

$32 = 100000$

$0.2 = 0.2 \times 2 = 0.4 \quad 0 \quad MSB$

$= 0.4 \times 2 = 0.8 \quad 0$

$= 0.8 \times 2 = 1.6 \quad 1$

$= 0.6 \times 2 = 1.2 \quad 1$

$= 0.2 \times 2 = 0.4 \quad 0$

$= 0.4 \times 2 = 0.8 \quad 0$

$= 0.8 \times 2 = 1.6 \quad 1$

$= 0.6 \times 2 = 1.2 \quad 1$

So a binary representation of $0.2$is given by 0.001100110011

64.2 is 100000.001100110011

Step 2: we normalize the binary representation

1000000.001100110011 Becomes $1.00000001100110011 \times 2^5$

Step 3: 5 is the true exponent.
In excess-16, we have $5 + 16 = 21$
21 in 5-bit unsigned representation is 10101

Step 4: The mantissa stored is what is on the right side of the radix point of the normal form.
So M is $000000011001100110011 0$

Putting all these together we have 64.2 represented as:

$$0\ \ 10101\ \ 00000001[10011001100110]$$

NB: On 8 bits we see that part of the number will be missing. That is why we say the mantissa determines the precision (exactness) of the number.

b. -15.375

Negative number ⇒ sign bit is 1

$15 = 1111$
$0.375 \times 2 = 0.75 \quad 0 \quad MSB$
$0.75 \times 2 = 1.5 \quad 1$
$0.5 \times 2 = 0.96 \quad 0$
$0.5 \times 2 = 1.0 \quad 1$

$0.375 = 0101$

$15.375 = 1111.0101$

Normalized we have, 1.1110101×2^3
$E = 3 \Rightarrow 19$in excess-16
$E = 10010$
$-15.375$is represented as 1 10010 11101010

c. 0.125

Positive number ⇒Sign bit is 0

$0 = 0000$
$0.125 \times 2 = 0.25 \quad 0 \quad MSB$
$0.25 \times 2 = 0.5 \quad 0$
$0.5 \times 2 = 1.0 \quad 1$

$0.125 = 0.001$
Normalized we have, 1.0×2^{-3}
$E = -3 + 16 = 13 = 01101$

0.125 is represented as 0 01100 00000000

2. Suppose the following 14-bit floating-point representation patterns, what decimal numbers do they represent?

a. 01000011000000 b. 10111010000000 c. 11111000000001

Solutions:

a. 010000110 00000

Sign bit is $S = 0$ ⇒positive number

$E = 10000 = 16$ (in normalized form)

Fraction is 1.11 (with an implicit leading 1) $= 1 + 1 \times 2^{-1} + 1 \times 2^{-2} = 1.75$

The number is $+1.75 \times 2^{(16-16)} = +1.75 \times 2^{0} = +1.75$

b. 10111010000000

Sign bit $S = 1$ ⇒ negative number

$E = 01110 = 13$ (in normalized form)

Fraction is 1.1 (with an implicit leading 1) $= 1 + 1 \times 2^{-1} = 1.5$

The number is $-1.5 \times 2^{(13-16)} = -1.5 \times 2^{-3} = -1.625$

c. 1 11110 00000001

Sign bit $S = 1$ ⇒ negative number

$E = 11110 = 30$ (in normalized form)

Fraction is 1.00000001 (with an implicit leading 1) $= 1 + 2^{-8}$

The number is $-(1 + 2^{-8}) \times 2^{(30-16)}$

6.6.4 Representation of Characters

Characters are represented using a chosen character encoding scheme (character set, charset, character map, or code page). The most commonly-used character encoding schemes are: ASCII, Latin-x for western European characters, and Unicode for internationalization.

6.6.4.1 ASCII Charset

ASCII stands for American Standard Code for Information Interchange. The ASCII character code is a 7-bit code used to represent numeric, alphabetic, and special printable characters. It also includes codes for *control characters*, which are not printed or displayed but specify some control function. A 7-bit means a total of $2^7 = 128$ characters can be represented.

- Code numbers 0 to 31 and 127 are special control characters, which are non-printable (non-displayable)
- Code numbers 65 to 90 represent 'A' to 'Z', respectively.
- Code numbers 97 to 122 represent 'a' to 'z', respectively.
  Code numbers 48 to 57 represent '0' to '9', respectively.

6.6.4.2 Latin-1

Latin alphabet No. 1 or Latin-1 in short, is the most commonly-used encoding scheme for western European languages. It has 191 printable characters from the Latin script, which covers languages like English, German, Italian, Portuguese and Spanish. Latin-1 is backward compatible with the 7-bit ASCII code. That is, the first 128 characters in Latin-1 (code numbers 0 to 127 (7FH)), is the same as ASCII. Code numbers 128 (80H) to 159 (9FH) are

not assigned. Code numbers 160 (A0H) to 255 (FFH) represent other special characters like Å, Æ, Ç, ~, £, ¤, ¥, § and ©.

6.6.4.3 Unicode

Unicode originally uses 16 bits (called UCS-2 or Unicode Character Set - 2 byte), which can represent up to 65,536 characters. Before Unicode, no single character encoding scheme could represent characters in all languages. Unicode is backward compatible with the 7-bit ASCII and 8-bit Latin-1. That is, the first 128 characters are the same as ASCII and the first 256 characters are the same as Latin-1.

The original 16-bit range of U+0000H to U+FFFFH (65536 characters) is known as *Basic Multilingual Plane* (BMP), covering all the major languages in use currently. The characters outside BMP are called *Supplementary Characters*, which are not frequently-used.

UCS-4 (Universal Character Set - 4 Byte), uses 4 bytes (32 bits), covering BMP and the supplementary characters.

CHAPTER 7: DIGITAL ELECTRONICS

7.1 BOOLEAN:

The Boolean data type has only two values: *true* or *false*. These values are used to control the flow of the execution of programs. Boolean values are found by comparing other data values. The results of these comparisons may be combined in Boolean expressions.

7.2 LOGIC GATES:

Many electronic circuits have to make decisions. They look at two or more **inputs** and use these to determine the **outputs** from the circuit. The process of doing this uses electronic logic, which is based on digital switches called gates.

Logic gates allow an electronic system to make a decision based on a number of its inputs. They are digital electronic devices. Each input and output of the gates must be one of two states:

- true or 1 or 'on'
- false or 0 or 'off'

A single digital signal can be either on or off - for example, a light with one switch can be on or off. However, if there is more than one signal, there are more than two possible states. For example, if two signals are present there are four possible combinations: on/on, on/off, off/on and off/off.

In a logic gate, each combination can be made to produce a different outcome. Binary numbers *reflect* the two states - on and off, 1 and 0, true and false - within CPUtransistors. Logic gate calculations can also be represented as truth tables.

7.3 BOOLEAN ALGEBRA

Boolean algebra and truth tables can be used to describe **logical expressions**. The most common Boolean operators are **AND**, **OR** and **NOT** (*always in capitals*). Each operator has a standard symbol that can be used when drawing logic gate circuits.

Boolean algebra also known as the algebra of logic was developed by an English mathematician called George Boole. It deals with binary variables and logic operators operating on these variables. A binary variable has only two possible values 0 or 1. Logic operators operating on these variables are AND (·), OR (+) and NOT (‘). Operations are defined for the values 0 and 1 as follows:

AND	OR	NOT
$0 \cdot 0 = 0$	$0 + 0 = 0$	$\bar{0} = 1$
$0 \cdot 1 = 0$	$0 + 1 = 1$	$\bar{1} = 1$
$1 \cdot 0 = 0$	$1 + 0 = 1$	
$1 \cdot 1 = 1$	$1 + 1 = 1$	

7.3.1 NOT GATE

A NOT gate has just one input. The output of the circuit will be the opposite of the input. If 0 is input, then the output is 1. If 1 is input, then 0 is output.

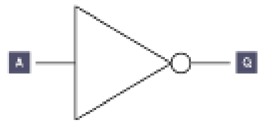


Figure 7.1: *NOT gate*

If A is the input and Q is the output, the truth table looks like this:

A	Q
0	1
1	0

Table 7.1: *Truth table for a NOT gate*

The Boolean expression is written as **Q = NOT A**.

7.3.2 AND gate

An AND gate can be used on a gate with two inputs. AND tells us that both inputs have to be 1 in order for the output to be 1.

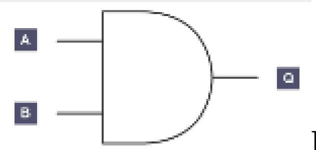


Figure 7.2: *AND gate*

The truth table would look like this:

A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

Table 7.2: *Truth table of an AND gate*

The Boolean expression is written as **Q = A AND B**.

7.3.3 OR GATE:

The OR gate has two inputs. One or both inputs must be 1 to output 1, otherwise it outputs 0.

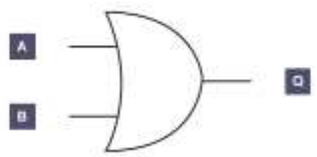


Figure 7.3: *OR gate*

The truth table would look like this:

A	B	Q

0	0	0
0	1	1
1	0	1
1	1	1

Table 7.3: *Truth table OR gate*

The Boolean expression is written as **Q = A OR B**.

7.3.4 XOR GATE:

The exclusive OR gate works the same as an OR gate, but will output 1 only if one or the other (not both) inputs are 1.

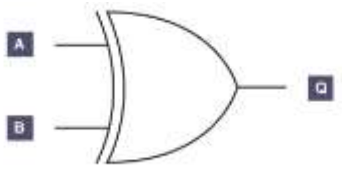


Figure 7.4: *XOR gate*

The XOR gate is indicated with the extra curved line to the left of the main shape. The truth table would read like this:

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

Table 7.4: *truth table of XOR gate*

The Boolean expression is written as **Q = A XOR B**.

7.3.5 NAND Gate

The NAND gate operates as an AND gate followed by a NOT gate. The output is "false" if both inputs are "true." Otherwise, the output is "true.

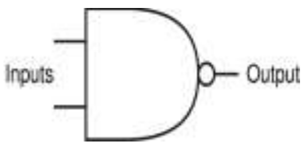


Figure 7.6: *NAND gate*

A	B	Output
0	0	1
0	1	1
1	0	1
1	1	0

Table 7.6: *Truth table of a NAND gate with two inputs*

7.3.6 NOR Gate

A NOR gate is equivalent to an OR gate followed by a NOT gate. Its output is "true" if both inputs are "false." Otherwise, the output is "false."

A	B	Output
0	0	1

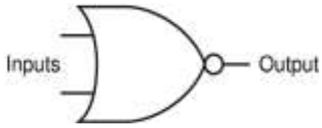


Figure 7.7: *NOR gate*

0	1	0
1	0	0
1	1	0

Table 7.7: *Truth table of a NOR gate with two inputs*

7.4 Complex Logic Gates:

Logic gates can be built up into chains of *logical decisions*. Some logic gates may have more than two inputs. The diagram below shows a *complex logic gate* combining three simple gates.

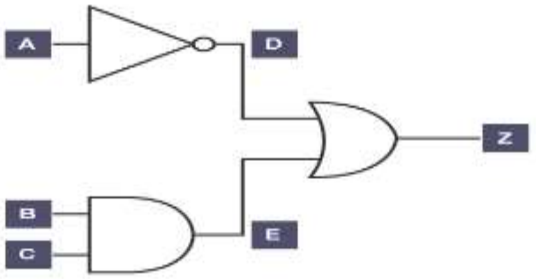


Figure 7.8: *complex logic gate, example 1*

Altogether there are three inputs and eight possible outcomes. To solve the truth table below, first find D, then E and finally Z. Complete a whole column before moving on to the next column. D depends only on A, E depends on B and C, and Z depends on E or D.

This logic gate truth table is written as:

A	B	C	D = NOT A	E = B AND C	Z = D OR E
0	0	0	1	0	1
0	0	1	1	0	1
0	1	0	1	0	1
0	1	1	1	1	1
1	0	0	0	0	0
1	0	1	0	0	0
1	1	0	0	0	0
1	1	1	0	1	1

Table 7.8: *truth table of the complex circuit in **figure 7.5***

This circuit would be written as **Z = D OR E** or **Z = NOT A OR (B AND C)**.

Logic Gates in the CPU

The following example demonstrates how the ALU uses logic gates to perform binary addition. It combines two gates, in parallel. There are two inputs and each gate has a single output - so in total there are two outputs, with four possible outcomes.

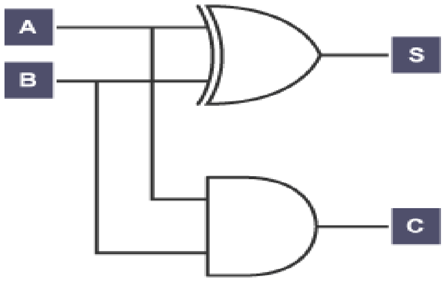


Figure 7.8: *Complex logic gate, example 2*

The Boolean expressions for this circuit are: **S = A XOR B** **C = A AND B**

The truth table for this circuit is:

A	B	S = A XOR B	C = A AND B
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Table 7.8: *Truth table of the circuit in **figure 7.6***

REVIEW:

(i) To convert a gate circuit to a Boolean expression, label each gate output with a Boolean sub-expression corresponding to the gates' input signals, until a final expression is reached at the last gate.

(ii) To convert a Boolean expression to a gate circuit, evaluate the expression using standard order of operations: multiplication before addition, and operations within parentheses before anything else.

7.5 Boolean Algebraic Identities:

Basic Boolean algebraic identities

Additive	Multiplicative
$A + 0 = A$	$0A = 0$
$A + 1 = 1$	$1A = A$
$A + A = A$	$AA = A$
$A + \overline{A} = 1$	$A\overline{A} = 0$

Basic Boolean algebraic properties

Additive	Multiplicative
$A + B = B + A$	$AB = BA$
$A + (B + C) = (A + B) + C$	$A(BC) = (AB)C$

$$A(B + C) = AB + AC$$

Useful Boolean rules for simplification

$$A + AB = A$$

$$A + \overline{A}B = A + B$$

$$(A + B)(A + C) = A + BC$$

7.6 DeMorgan's Theorems

DeMorgan, a mathematician who knew Boole, proposed two theorems that are an important part of Boolean algebra. In practical terms, DeMorgan's theorems provide mathematical verification of the equivalency of the NAND and negative-OR gates and the equivalency of the NOR and negative-AND gates.

One of DeMorgan's theorems is stated as follows:

The complement of a product of variables is equal to the sum of the complements of the variables,

Stated another way,

The complement of two or more ANDed variables is equivalent to the OR of the complements of the individual variables.

The formula for expressing this theorem for two variables is:

$\overline{XY} = \overline{X} + \overline{Y}$

DeMorgan's second theorem is stated as follows:

The complement of a sum of variables is equal to the product of the complements of the variables.

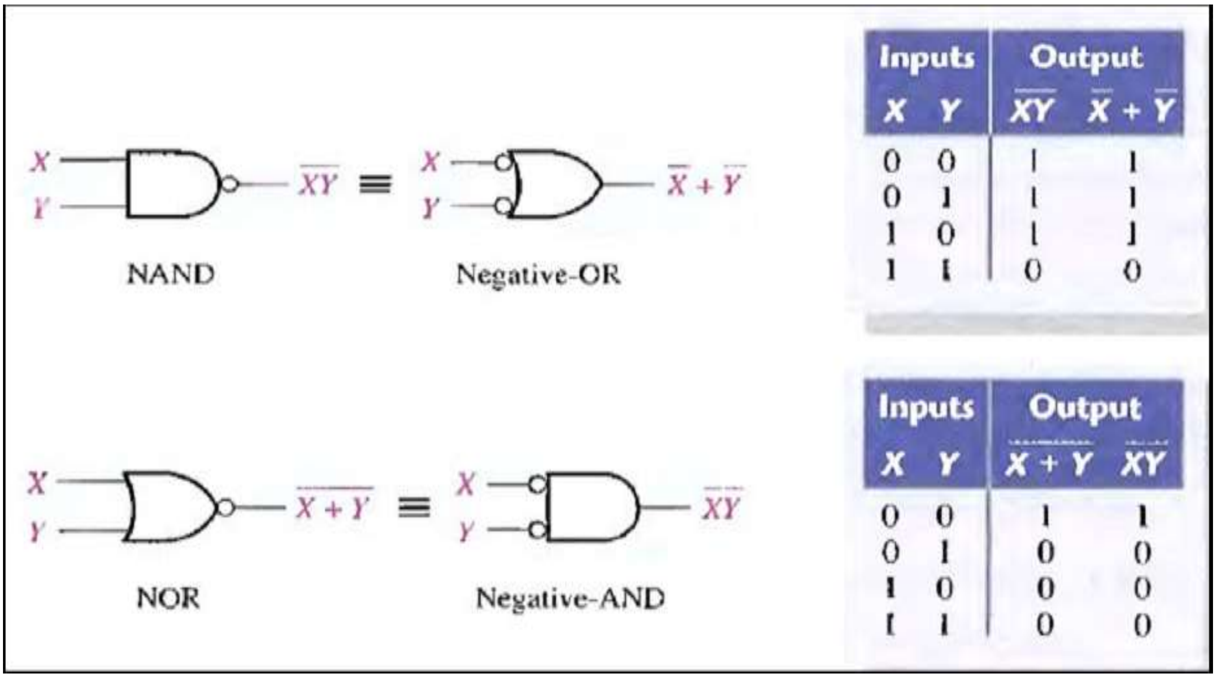
Stated another way**,**

The complement of two or more ORed variables is equivalent to the AND of the complements of the individual variables,

The formula for expressing this theorem for two variables is

$\overline{X + Y} = \overline{X}\overline{Y}$

The figure below shows the gate equivalencies and truth tables for the two equations above



Inputs		Output	
X	Y	$\overline{XY}$	$\overline{X} + \overline{Y}$
0	0	1	1
0	1	1	1
1	0	1	1
1	1	0	0

Inputs		Output	
X	Y	$\overline{X + Y}$	$\overline{X}\,\overline{Y}$
0	0	1	1
0	1	0	0
1	0	0	0
1	1	0	0

Figure 7.9: Gate equivalencies and the corresponding truth tables thatillustrate DeMorgan's theorems

7.7 Simplifying Boolean Expressions

Simplifying a Boolean expression consists of using the laws of Boolean algebra to write the expression in its simplest form.

Example 1: Simplify the expression $AB + A(B + C) + B(B + C)$
Solution:

$AB + A(B + C) + B(B + C) = AB + AB + AC + BB + BC$
$= AB + AB + AC + B + BC$ $\quad (BB = B)$
$= AB + AC + B + BC$ $\quad (AB + AB = AB)$
$= AB + AC + B$ $\quad (B + BC = B)$
$= B + AC$ $\quad (AB + B = B)$

Example 2: Simplify the expression
$A\bar{B} + A(\overline{B + C}) + B(\overline{B + C}) = A\bar{B} + A(\bar{B}\bar{C}) + B(\bar{B}\bar{C})$ (De Morgan)
$= A\bar{B} + A\bar{B}\bar{C} + B\bar{B}\bar{C}$
$= A\bar{B} + A\bar{B}\bar{C} + 0\bar{C}$ $\quad (B\bar{B} = 0)$
$= A\bar{B} + A\bar{B}\bar{C} + 0$ $\quad (0C' = 0)$
$= A\bar{B}(1 + C)$ $\quad (1 + C' = 1)$
$= A\bar{B} \cdot 1$ $\quad (A\bar{B} \cdot 1 = A\bar{B})$
$= A\bar{B}$ $\quad (A\bar{B} \cdot 1)$

Example 3: Apply DeMorgan's theorems to the expressions $\overline{XYZ}$ and $\overline{X + Y + z}$.
$\overline{XYZ} = \bar{X} + \bar{Y} + \bar{Z}$
$\overline{X + Y + z}. = \bar{X}\bar{Y}\bar{Z}$

Drill Questions:

1: Construct logic circuits for the following expressions

a) $\bar{A}B + A\bar{B}$ (b) $(A + B)(\overline{A + \bar{B}})$ (c) $\bar{A}(\bar{B} + A)$ (d) $(\bar{A} + B)(\overline{\mathrm{ABC}})$ (e) $(A + \bar{B})AC$

2: Show that

$(a) A + AB = A$ $(b) (\overline{A + B}) = \bar{B}(\bar{A} + B)$ (c) $(A + B)(A + \bar{B}) = A$

3: Use De Morgan's theorems to simplify the following

i) $(\overline{A + BC + CB})$

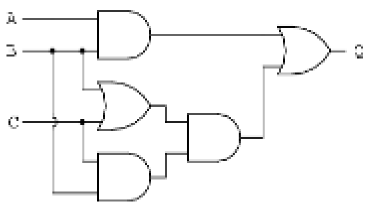
ii) $(\overline{A\bar{B} + A\bar{C}})$

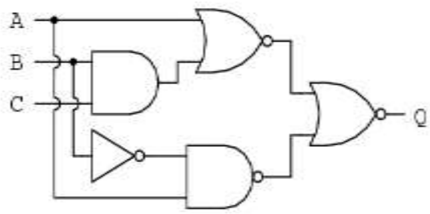
iii) $\overline{(A + B + C)D}$

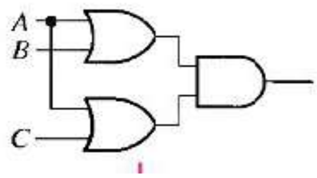
iv) $\overline{ABC + DEF}$

v) $\overline{A\bar{B} + C\bar{D} + EF}$

4: What are the outputs for the following logic circuits,







Chapter 8: INFORMATION SYSTEMS

8.1 Introduction

A system is an integrated set of regularly interacting or interdependent components created to accomplish a defined objective, with defined and maintained relationships among its components. Basically, there are three major units in every system namely input, processing and output. The objective of a system demands that some output be produced as a result of processing the suitable inputs.
An information system can therefore be seen as a set of interrelated components that collect data, process the data to produce information.

8.2 Information System Components

There are six basic components in an information system: hardware, software, procedures, data and people.

a) Hardware: Hardware refers to the physical devices that make up the system. They are the whole set of equipment used for input, processing, storage and communication of data.

b) Software: Software is the collection of computer programs used in the system. They provide the instructions that tell the computer what to do.

c) Data: Data are raw, unorganized, potentially useful facts and figures that are processed to produce information.

d) People: People are the main actors of the system. They are the users of the information system. They input data into the computer, give some direction to the computer to perform tasks and review information on the computer for output.

e) Procedures: Procedures are the series of documented actions taken to achieve a particular goal. A procedure is more than a single simple task. It can be complex and involved, such as reinstalling software, performing a backup etc.

8.3 Organizational Information Systems

There are three levels at which information can be used in an organisation: ***strategic***, ***tactical***and ***operational***levels. This can be represented using the pyramid below.

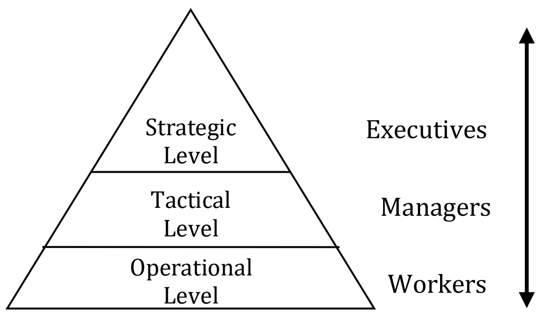


Figure 8.1: *Representation of information in an organisation*

a) At the **strategic level**, information is needed by senior managers (executives) to help them with their business plans. Information at this level is used for making long term decisions.
b) At the **tactical level**, information is needed by middle managers to help them monitor and control business activities. Tactical planning and decision-making takes place within the guidelines set by the strategic plan.
c) At the **operation level**, employees with operational roles need information to help them carry out their duties. Results of operational work are passed upwards to let the tactical planners evaluate their plans.

8.4 Types of Information systems

In order to meet with the information needs of the organization, different types of information systems exist which can be grouped into two: **operations support systems** and **management support systems**.

8.4.1 Operations Support Systems

Operation support systems process data generated by business operations. They act at the operational level of the organization. Major categories of OSS are **transaction processing systems**, **office automation systems** and **process control systems**.

8.4.1.1 Transaction Processing Systems

A transaction is any event of interest to an organization. It may be a business activity such as a payment, a deposit, a customer's order, a reservation or a student's registration. Transaction processing systems capture and process data generated during an organization's day-to-day transactions and maintain records about the transactions. They are vital for any organization or business as they gather all the input necessary for other types of systems. TPS are also called **Data Processing Systems**.

There are two types of TPS: ***batch processing*** and ***online processing systems***.

a) Batch Processing
With batch processing, transaction data is collected over a period of time and all processing is done as a group. Batch processing is ideal in situations where large amounts of data requiring similar processing are to be processed. Examples are:

- **Payroll systems** for calculating employee salaries
- **Billing systems** for calculating consumer bills.

b) Online Transaction Processing
With online transaction processing (OLTP) the computer processes transactions as they are entered. Such systems are ideal for situations where the master file needs to be updated each time a transaction is made. Examples are:

- **Stock control systems** which reduce automatically the number of items in stock once an item has been bought
- **Reservation systems** which reduce automatically the number of seats available on a flight or bus once a seat has been booked.

8.4.1.2 Office Automation Systems

Office automation systems automate office procedures and enhance office communication and productivity. They support a wide range of office activities such as creating and distributing documents, sending messages and scheduling. The software an OAS uses to support these activities include word processing, spreadsheets, databases, presentation, graphics, e-mail, Web browsers, personal information management, and groupware. They use communication technologies such as voice mail, facsimile (fax), videoconferencing, and electronic data interchange (EDI) for the electronic exchange of text, graphics, audio, and video. OAS are also called **Office Information Systems** (OIS).

8.4.2 Management Support Systems

Management support systems provide information and support needed for effective decision making by managers. They act at the tactical and strategic levels of the organization. Major categories of MSS are management information systems, decision support systems and executive information systems.

8.4.2.1 Management Information systems

Management information systems generate accurate, timely and organized information needed by middle managers to take decisions, solve problems, supervise activities, and track progress. They provide routine information for routine tasks. The source of data for an MIS usually comes from numerous databases. These databases are usually the data storage for Transaction Processing Systems. MIS take information from TPS and summarize them into a series of management reports. As such, MIS are sometimes called **Management Reporting Systems** (MRS).
MIS generate three basic types of information or reports: detailed, summary and exception.

i) **Detailed reports** confirm transaction processing activities. A detailed order report is an example of a detail report.

ii) **Summary reports** consolidate data into a format that an individual can review quickly and easily. To help synopsize information, a summary report typically contains totals, tables, or graphs. An inventory summary report is an example of a summary report.

iii) **Exception reports** report information that is outside of a normal condition. These conditions called the exception criteria, define the range of what is considered normal activity or status. An example of an exception report is an inventory exception report that notifies the purchasing department of items it needs to reorder. Exception reports help managers save time because they do not have to search through a detailed report for exceptions. Instead, an exception report brings exceptions to the manager’s attention in an easily identifiable form. Exception reports thus help them focus on situations that require immediate decisions or actions.

Examples of MIS are:

- ✓ Sales management systems
- ✓ Inventory control systems

- ✓ Budgeting systems
- ✓ Management reporting systems

8.4.2.2 Decision Support Systems

Decision support systems are designed to help tactical and strategic decision-making in situations where there is uncertainty about the possible outcomes of those decisions. They provide interactive support for non-routine decisions or problems.

TPS and MIS provide information on a regular basis. However, managers need information not provided in their reports to help them make decisions. Decision support systems therefore use data from internal (TPS and MIS) and external sources.

- o Internal sources of data might include sales, manufacturing, inventory, or financial data from an organization's database.
- o Data from external sources could include interest rates, population trends, and costs of new housing construction or raw material pricing.

Examples of DSS are:

- ✓ Logistics Systems
- ✓ Financial Planning Systems
- ✓ Spreadsheet Models

8.4.2.3 Executive Information Systems

Executive information systems (EIS) are designed to support the information needs of executive management. Their purpose is to analyse, compare and identify trends to help the strategic direction of the organisation. Information in an EIS is presented in charts and tables that show trends, ratios, and other managerial statistics. Because executives usually focus on strategic issues, EISs rely on external data sources that can provide current information on interest rates, commodity prices, and other leading economic indicators.

To store all the necessary decision-making data, DSSs or EISs often use extremely large databases, called data warehouses.

8.4.3 Other Information Systems

8.4.3.1 Expert Systems

An expert system is a computer program that tries to emulate the decision making of a human expert. It does this by combining the knowledge of human experts and then, following a set of rules, it draws inferences. An expert system is made up of three parts: a **knowledge base**, an **inference engine** and a **user interface**.

i. The **knowledge base**stores all of the facts, rules and information needed to represent the knowledge of the expert.

ii. The **inference engine**is the part of the system that interprets the rules and facts using backward and forward chaining to find solutions to user queries.

iii. The **user interface**allows the user to enter new knowledge and query the system.

Example 1:

A medical diagnosis expert system could be used in a doctor's waiting room. Patients would use a touch screen to answer questions on symptoms etc. created by the system. Based on the

patient responses, the system could use its database of diseases and symptoms, along with its programmed rules, to prepare a list of possible diagnosis for the doctor to investigate further.

Advantages

- The doctor saves time because they do not have to ask the patient to describe their symptoms in person.
- The doctor is given a suggested list of possible diagnosis to investigate further.
- The computer can store far more information than the doctor and can search it far faster and more efficiently.
- The database can easily be updated or extended.

Disadvantages

- It can be difficult to describe symptoms to a computer system.
- It relies on a basic level of skills from the user.
- It lacks the 'human touch' of a doctor actually talking to a patient.

Example 2:

Aexpert system used by a car mechanic could help to diagnose faults in a car by asking the mechanic to carry out tests or answer questions. This could also be automated as the computer could have inputs from sensors or have a direct interface with a computer system built into the car. The system would then give the mechanic a list of probable faults with the car even make automatic adjustments using the computer system built into the car. It would also be helpful to a newly qualified mechanic as they would have access to a wider range of knowledge, could save time compared to having to contact an expert and the system could be used as a training aid.

Advantages

- The user saves time because the mechanic is taken through a logical series of things to try out that are likely to solve common problems.
- The system can directly access computer systems built into many modern cars.
- The system can store details of a huge range of common faults with different makes of cars.
- The database can easily be updated or extended as new problems are identified.

Disadvantages

- It can be difficult to answer questions on something the user may know nothing about.
- It relies on a basic level of skills from the user.

Expert systems are one part of an exciting branch of computer science called **artificial intelligence (AI)**. AI is the application of human intelligence to computers. AI technology can sense your actions and, based on logical assumptions and prior experience, will take the appropriate action to complete the task. AI has a variety of capabilities, including *speech recognition, logical reasoning,* and *creative responses*.

8.4.3.2 Geographical Information Systems

A geographic information system (GIS) is a computer system for capturing, storing, checking, and displaying data related to positions on Earth's surface. GIS can show many different kinds of data on one map. This enables people to more easily see, analyse, and understand patterns and relationships.

With GIS technology, people can compare the locations of different things in order to discover how they relate to each other. For example, using GIS, the same map could include sites that produce pollution, such as gas stations, and sites that are sensitive to pollution, such as wetlands. Such a map would help people determine which wetlands are most at risk.

GIS can use any information that includes location. The location can be expressed in many different ways, such as latitude and longitude, address, or ZIP code. Many different types of information can be compared and contrasted using GIS. The system can include data about people, such as population, income, or education level. It can include information about the land, such as the location of streams, different kinds of vegetation, and different kinds of soil. It can include information about the sites of factories, farms, and schools, or storm drains, roads, and electric power lines.

8.4.3.3 Health Information Systems

Health information systems refer to any system that captures, stores, manages or transmits information related to the health of individuals or the activities of organisations that work within the health sector. This definition incorporates things such as district level routine information systems, disease surveillance systems, and also includes laboratory information systems, hospital patient administration systems (PAS) and human resource management information systems (HRMIS). Overall, a well-functioning HIS is an integrated effort to collect, process, report and use health information and knowledge to influence policy and decision-making, programme action, individual and public health outcomes, and research. Sound decision-making at all levels of a health system requires reliable health statistics that are disaggregated by sex, age and socioeconomic characteristics. At a policy level, decisions informed by evidence contribute to more efficient resource allocation and, at the delivery level, information about the quality and effectiveness of services can contribute to better outcomes.

Information systems, particularly at lower levels of the health system (*closer to the collection source*), need to be simple and sustainable and not overburden health delivery staff or be too costly to run. Staff need feedback on how the routine data they collect can be used and also need to understand the importance of good quality data for improving health. Capacity building is required to ensure policymakers at all levels have the ability to use and interpret health data, whether it originates from routine systems, health surveys or special operational research. It is also important that health system staff understand the significance of local data for local program management, and that their needs for strengthened capacity for critical health statistical analysis are met. Local use of data collected at lower levels of the health system is a key step for improving overall data quality. Furthermore, aggregate patient information collected at various points of service delivery and made interoperable with routine HIS improves the quality and use of health information.

The Health Metrics Network (HMN), in their *Framework and Standards for Country Health Information Systems*, has defined a Health Information System as consisting of six components:

Figure 8.2: *Health Metrics Network*

i) **Health Information Systems Resources:** These include the legislative, regulatory and planning frameworks required for a fully functioning health information system, and the resources that are required for such a system to be functional. Such resources involve personnel, financing, logistics support, information and communications technology (ICT), and coordinating mechanisms within and between the six components

ii) **Indicators**: A core set of indicators and related targets is the basis for a health information system plan and strategy. Indicators need to encompass determinants of health; health system inputs, outputs and outcomes; and health status

iii) **Data Sources:** These can be divided into two main categories; (1) population-based approaches (censuses, civil registration and population surveys) and (2) institution-based data (individual records, service records and resource records). A number of data-collection approaches and sources do not fit into either of the above main categories but can provide important information that may not be available elsewhere. These include occasional health surveys, research, and information produced by community based organisations

iv) **Data Management**: This covers all aspects of data handling from collection, storage, quality-assurance and flow, to processing, compilation and analysis

v) **Information Products:** Data must be transformed into information that will become the basis for evidence and knowledge to shape health action

vi) **Dissemination and Use:** The value of health information is enhanced by making it readily accessible to decision-makers and by providing incentives for, or otherwise facilitating, information use.

8.5 Architectural Requirements of an Information System

8.5.1 Data Vs Information

Data are simply facts or figures — *bits* of information, but not information itself. When data are processed, interpreted, organized, structured or presented so as to make them meaningful or useful, they are called *information*. Information provides context for data.

For example, a list of dates — data — is meaningless without the information that makes the dates relevant (dates of holiday).

Examples of Data and Information.

- The history of temperature readings all over the world for the past 100 years is data. If this data is organized and analysed to find that global temperature is rising, then that is information.

- The number of visitors to a website by country is an example of data. Finding out that traffic from the U.S. is increasing while that from Australia is decreasing is meaningful information.

Data is processed by computers, the resulting information can then be used to form judgements and make predictions.

Input devices can collect data automatically, e.gsensors that continually measure a temperature or a fix-mount barcode reader at a till.

In both of these cases the data collected will be read into a database for processing. With a structure in place (the database) the data becomes information.

Spreadsheets are commonly used to turn data into information.

8.5.2 Knowledge:

Knowledge is the ability to understand information and to then form judgements, opinions, make predictions and decisions based on that understanding.

Knowledge from information

Example 1: Each year, for the past five years, the apple crop in Somerset has grown by 10%. The same amount of growth is predicted this year so we need to find markets for a further 10% of apples.

In this example, data collected each year, for the past five years, has become information and a pattern in the growth of the apple crop has been identified - 10% year on year. This information has been used to predict the same level of growth this year and has highlighted the need for more markets. The prediction and realisation are ***knowledge***, ie the use of information.

Example 2: The car looked like it had flat tyres and it had a pool of petrol underneath it at the back. There was smoke coming from the bonnet. I took the decision not to allow anyone near it and to evacuate the area.

In this example the judgment made and the decision that followed, ie to evacuate the area, is the **knowledge**, gained after assessing the available **information**.

Data leads to information, and information leads to knowledge.

8.5.3 Sources of Data

a) Questionnaire: A questionnaire is a set of questions used for collecting data from people. A questionnaire may be in paper format or online.

b) Interview: An interview is a meeting during which somebody is asked questions. Interviews allow you to collect a greater depth of data and understanding from people than is possible by just using a questionnaire.

c) Observation: In observation, the data gatherer observes what is happening during a process or event and produces some kind of data file as a result

d) Data Logging: Data logging is an automated method of gathering data by using sensors.

e) Document Review: Document review is getting relevant data from a document, an article or a book.

f) Data Mining: Data mining is the exploration of databases to collect data.

8.5.4 Characteristics of Information

Good information is that which is used and which creates value. Experience and research show that good information has numerous qualities.

a) Timeliness: Delay destroys the value of information. For effective decision making, information must reach the decision-maker at the right time. Timeliness means that information must reach its recipients within the prescribed timeframes.

b) Accuracy: Wrong information given to decision-makers would result in wrong decisions. Accuracy means that information should be free from mistakes and errors.

c) Current: For the characteristic of timeliness to be effective, information should be current or up-to-date. Information must be current as a fact of yesterday may not be a fact of today.

d) Completeness: Information should have every necessary part or everything that is wanted. If information is not complete, it may lead to wrong decisions being made as only half of an entirety of the information is known.

e) Explicitness: Good information should not require further analysis for decision making. It should be clear and obvious, leaving no doubts as to its intended meaning.

8.5.5 Data Collection

All computer systems need to have data input into them otherwise they have nothing to process. Getting the data for the computer to process is known as data collection. Data collection can be manual or automatic.

8.5.5.1 Manual Data Collection

Manual data collection uses forms and questionnaires. Data collected through this method has to be entered into the computer by typing and clicking.
Many different errors can occur when entering data into a system. To try and reduce the amount of input errors, a system designer can build in validation and verification checks into the software that the data is entered into.

i) Validation: Validation is an automatic computer check to ensure that the data entered is sensible and reasonable. It does not check the accuracy of data.

For example, a secondary school student is likely to be aged between 11 and 16. The computer can be programmed only to accept numbers between 11 and 16. This is a **range** check.

However, this does not guarantee that the number typed in is correct. For example, a student's age might be 14, but if 11 is entered it will be valid but incorrect.

Figure 8.3: *Illustration of datavalidation using range check*

a) Types of validation

There are a number of validation types that can be used to check the data that is being entered.

Validation type	How it works	Example usage
Check digit	The last one or two digits in a code are used to check the other digits are correct	Bar code readers in supermarkets use check digits
Format check	Checks the data is in the right format	A National Insurance number is in the form LL 99 99 99 L where L is any letter and 9 is any number

Table 8.1: *types of data validation, functions and examples of usage*

Length check	Checks the data isn't too short or too long	A password which needs to be six letters long
Lookup table	Looks up acceptable values in a table	There are only seven possible days of the week
Presence check	Checks that data has been entered into a field	In most databases a key field cannot be left blank
Range check	Checks that a value falls within the specified range	Number of hours worked must be less than 50 and more than 0
Spell check	Looks up words in a dictionary	When word processing

Check digits: A check digit is a digit attached to the end of a string of digits that can be used to check that the string is correct. It is calculated from the other digits in the string. One example where a check digit is used is in the 10 digit ISBN number which uniquely identifies books. The last number of the ISBN is actually the check digit for the other numbers.
For example, in the ISBN 185813415**3**, the 3 at the end of the number is the check digit**.**
The check digit for ISBNs is obtained using a calculation method known as the Modulus-11 weighted check digit calculation.

- Start with original number i.e. 185813415
- Weight each digit by its position in the string and add up the results.

Position	10	9	8	7	6	5	4	3	2	1
Digit	1	8	5	8	1	3	4	1	5	
Weightings	10	72	40	56	6	15	16	3	10	

Table 8.2: *check digits application using an ISBN (a)*

Total = 10+72+40+56+6+15+16+3+10= 228

- Divide the total by 11 and then subtract the remainder from 11. The check digit is the result of this operation.
  228 / 11 = 20 remainder 8 => Check digit is 11-8 = 3.
- Add the check digit to the end of the original number to get the complete product number. i.e. 185813415**3.**

To check whether the ISBN is correct,
- Input the number including the check digit.
- Weight each digit by its position in the string and add up the results.

Position	10	9	8	7	6	5	4	3	2	1
Digit	1	8	5	8	1	3	4	1	5	3
Weightings	10	72	40	56	6	15	16	3	10	3

Table 8.3: *check digits application using an ISBN (b)*

Total = 10+72+40+56+6+15+16+3+10+3 = 231

- Divide the total by 11. If the remainder is 0, then the number has passed the validation check and so it is likely that it has been inputted correctly.
  231 / 11 = 21 remainder 0

ii) Verification:
Verification is performed to ensure that the data entered exactly matches the original source.

There are two main methods of verification:

1. **Double entry** - entering the data twice and comparing the two copies. This effectively doubles the workload, and as most people are paid by the hour, it costs more too.
2. **Proofreading data** - this method involves someone checking the data entered against the original document. This is also time-consuming and costly.

8.5.5.2 Automatic Data Collection

Automatic data collection is a form of data input in which there is no data entry. It uses sensors and specialized input devices to collect data that is directly entered into the computer without any human involvement. It is also called data capture. Different automatic data collection methods are:

a) Optical Mark Recognition (OMR)
OMR uses a device called an optical mark reader to read marks made with prescribed pens, pencils or special writing material on OMR forms, and convert them into information in the computer. This system is good for multiple choice examination questions.

b) Optical Character Recognition (OCR)
This method uses a device called an optical character reader to read characters from printed or handwritten text and transmit them to the computer as if they were typed from the keyboard. This method is suitable for capturing data from airline tickets; reading postal codes; capturing data from telephone and electric bills.

c) Magnetic Ink Character Recognition (MICR)
The device used is a magnetic ink character reader that reads characters written in magnetic ink, using magnetic stripe readers or card swipe machines that capture the information on the magnetic card. These are seen on the back of credit cards and bank cards.

d) Barcode Reading
An optical device called barcode reader is used to read the barcode on products and convert them into a form that can be processed by the computer. A bar code is a sequence of vertical

lines and numbers that identify a product. They are used in libraries, supermarkets and retail shops.

e) Voice Recognition

This method converts speech into text or a sequence of computer commands. It is most common for data entry and word processing environments.

8.5.6 Data Security:

Data security is about keeping data safe and affects anyone relying on a computer system. If the data on a computer system is damaged, lost, or stolen, it can lead to disaster.

8.5.6.1 Key threats to data security

Data may get:

i. lost or damaged during a system crash - especially one affecting the hard disk
ii. corrupted as a result of faulty disks, disk drives, or power failures
iii. lost by accidentally deleting or overwriting files
iv. lost or become corrupted by computer viruses
v. hacked into by unauthorised users and deleted or altered
vi. destroyed by natural disasters, acts of terrorism, or war
vii. deleted or altered by employees wishing to make money or take revenge on their employer

8.5.6.2 Keeping data secure

Measures that can be taken to keep data secure include:

i. Making regular backups of files (backup copies should be stored in fireproof safes or in another building)
ii. Protecting yourself against viruses by running anti-virussoftware
iii. Using a system of passwords so that access to data is restricted
iv. Safe storage of important files stored on removable disks, eg locked away in a fireproof and waterproof safe
v. Allowing only authorised staff into certain computer areas, eg by controlling entry to these areas by means of id cards or magnetic swipe cards
vi. Always logging off or turning terminals off and if possible locking them
vii. Avoiding accidental deletion of files by write-protecting disks
viii. Using data encryption techniques to code data so that it makes no apparent sense

8.5.6.3 Online banking

When you bank online, after you've logged in, you will notice that the http in the address bar has changed to ***https***. This indicates that a secure connection between your computer and the bank's computer has been established. Data sent between the two computers is encrypted so that anyone trying to intercept your data will receive meaningless data. The data can only be

decrypted into readable data by using a key that is known only to the two computers - yours and the bank's.

8.5.7 Data Transfer:

Data transfer files have standards so they can be transferred to any computer, hardware or application. Graphics, text, sound or numbers don't need to be transferred on physical media anymore.

8.5.7.1 Data file types

Data files are stored in a number of formats, the formats depend on which application created the file to begin with. For example, Microsoft Word stores files as *.doc but Adobe Photoshop stores files in a different format - *.psd. The file extension identifies the file's format.

8.5.7.2 Opening files of different formats

When data is transferred from one computer to another, the computer receiving the data file may not be able to read the format without the right application installed. For example, an image created in Adobe Photoshop and saved as a *.*psd* file (Photoshop's format) would not be readable by Microsoft Paint.

This is becoming less of a problem as standardization matures and applications expand the list of file types they're able to read.

8.5.7.3 Translating between file types

Data can be stored temporarily in a computer's RAM as a means of transfer from one application to another. In Microsoft operating systems this feature is called the ***clipboard*** and it can ***copy*** data from one application and ***paste*** it into another. Nearly all other operating systems support this functionality.

For example you can create a graph using a spreadsheet application and then use the clipboard to copy the graph to a desktop publishing application.

8.5.7.4 Limitation

1. Certain elements of one document may not be copiable to another application, eg video.
2. Some file features (eg text layout and formatting) may be lost in the translation to a different format.

8.5.7.5 Import and export

Most applications have the ability to import and export data. Export saves the file in a format that's readable by other applications. Import opens a file created in another application for viewing or editing.

8.5.7.6 Standard file types

The need to import and export data files has led to the development of several standard file types that many applications can understand. Examples are **jpg** and **gif** files for images, and **mp3** files for sound; but there are also standard file types for text, movies, and spreadsheet data.

Often an application of a different type can import data, for example, a word processor may be able to import a spreadsheet file.

8.5.7.7 XML files

Recently program developers have started to use the XMLweb page file format as the standard way to store data, for example, the new Microsoft Office suite makes use of XML. RSS feeds on the internet also make use of this format. Such files can be read by any browser on any computer, making it very easy to transfer data between computers.

8.5.7.8 Other examples

Other standard file types, such as zip and pdf, have been developed as ways of distributing data in the most efficient way possible. They do this by compressing it using zip technology or making it possible to read using a freely available downloaded reader application, as is the case with PDF files.

8.5.7.9 Rapid transfer of data

Files can be easily transferred and shared across the globe using the internet. A file could be:

- emailed as an attachment
- sent over instant messenger
- downloaded from a website/web server
- accessed over a private network

Documents, eg a spreadsheet, can be stored on a web server and then accessed via any computer with an internet connection and a web browser. Collaborative working is possible too, where multiple people contribute to the same document.

This makes the world very small as far as transfer of data is concerned. It has allowed many companies to outsource some or all of their operations. This is where companies transfer activities such as ticket processing for airlines or telephone banking queries to workers in a country where wages and running costs are relatively low. Often the time difference between the two countries gives even greater convenience.

8.5.8 SYSTEMS

An organized, purposeful structure that consists of interrelated and interdependent elements (components, entities, factors, members, parts etc.). These elements continually influence one

another (directly or indirectly) to maintain their activity and the existence of the system, in order to achieve the goal of the system. All systems have;

(a) Inputs, outputs and feedbackmechanisms,

(b) Maintain an internal steady-state (called homeostasis) despite a changing external environment,

(c) Display properties that are different than the whole (called emergent properties) but are not possessed by any of the individual elements, and

(d) Have boundaries that are usually defined by the system observer. Systems underlie every phenomenon and all are part of a larger system. Systems stop functioning when an element is removed or changed significantly.

8.5.8.1 Natural and Artificial systems

a) Natural Systems

a. Such Systems exist and also abound in the nature.
b. Are also not at all the results of the human endeavors.
c. Rivers, mountains, minerals etc. are the major examples of the natural Systems.

b) Artificial Systems

a. Are manufactured (man made).
b. Examples of such Systems are dams, canals, roads, machines, factories etc.

8.6 DATABASE

Databases are used to organise data in a clear and consistent way. Most website and online applications use databases. With so much data now being shared online, data security is an important issue.

8.6.1 Uses of databases

Databases are very powerful tools used in all areas of computing. It is a key computing skill to be able to organize data, create databases and control data using query languages.

One of the main benefits of computer databases is that they make it easy to store information so it is quick and easy to find. For example, if you have music files on your computer, a media application like iTunes, Windows Media Player or Google Music organises that data so it is easy for you to quickly search for the artist or songs you want.

Most websites use databases to store data. Social networking sites use databases to store data about millions of users, along with photographs and other information about themselves and others.

Sport teams use data in performance analysis. Players and managers can use data to monitor levels of fitness and learn how to improve their skills.

Databases are also used to store data about weather patterns - which can then be used with software models to predict future events. Even data from internet search terms can be useful. For example, the Google Analytics database helped to successfully predict flu outbreaks around the world.

8.6.2 Database software

Database software includes off-the-shelf software such as **Microsoft Access**, **Libre Office Base**, **Oracle**, MySQL or NoSQL.

Databases can also be created and organized using programming languages. Languages like SQL, **Visual Basic** and **Delphi** are used to edit databases. Using programming languages means that you can customize a database to do exactly as you want.

8.6.3 Using data

There is so much data being captured online and through smartphones that data is being used in many areas of life. Big data, open data and data mining are important terms when working with data.

a) Big data

Big data describes extremely large sets of data. It includes data gathered from many different sources that is then analysed. Big data is often used for making predictions based on patterns that can be seen in the data.

b) Data mining

Data mining is a term used to describe analysing large amounts of data to predict future events and trends. As there is so much data now available, people who are able to analyse and understand data are going to be well placed to shape the development of technology.

c) Open data

Many organisations now share large sets of data freely. Organisations like the government, local councils, and world organisations (like the United Nations) make data freely available. This means that anyone can look at and analyse the data. Journalists often use open data sets to form the basis of news articles.

Open data projects are used to collaborate and share data around the world. For example, the Skynet project allows amateur and professional astronomers around the world to share information about the stars, planets, satellites and meteors.

d) Data security

We put a lot of trust in companies when we give them our personal details. They have a legal obligation to ensure that our data is kept in a secure centraliseddatabase. Data encryption is a necessity for any database containing personal data.

8.6.4 Designing databases

A database needs to be reliable, consistent and have a structure that suits the data you collect. A database is stored in a format such as a **CSV** *(Comma-separated values* - a standard file format for a flat-file database used in spreadsheet and database software) file saved on a hard drive.

*Databases are persistent, which means that the structure is fixed in place***.** The fields and data values are set so that it is easy to add information and build a database without changing the structure. If you create a program which needs to access a database, the database needs to be persistent so that records can be modified while the program runs.

When designing a database, it is important to decide what the structure will be before you start adding data. It is difficult to change a database structure once it has been set in place.

If data was not organised, it would be difficult to work with. For example, it is hard to make sense of this table which contains data for an address book:

Ash	Nia	becky@bbc.com	02398 374927
02298 837492	James	Irfan	Becky
james@bbc.com	04972 048204	Mobile	irfan@bbc.com
04972 048204	nia@bbc.com	Email	Parker

Table 8.4a & b: (a) *unorganized table with data for an address book; (b) organised table below*

It is easier for us to understand the data if it is presented with a clear layout - like it is in this table:

ID	Name	Home Phone	Mobile	Email
1	Irfan	02298 837492	0888 87492	irfan@bbc.com
2	James	02398 374927	098284 278213	james@bbc.com
3	Becky	03472 827492	098252 472911	becky@bbc.com

ID	Name	Home Phone	Mobile	Email
4	Nia	04972 048204	046482 472912	nia@bbc.com

Flat-file databases

A flat-file database is a database of just one table. It can be created in database software or in a spreadsheet and is often saved as a CSV file. A flat file is useful because it is highly compatible between databases and other applications.

Examples of flat-file databases: Flat-file databases could be used for a number of things, eg:

- usernames and passwords
- contact details
- product details
- game or music collections
- entities and attributes

8.6.5 Creating a database

When creating a database think about what data you need to store. A database is essentially a collection of details about different items. You could create a database about pretty much anything.

i) Entities

When you build a database you are organising data about entities. An entity is any item that has its attributes stored as data. An entity could be anything, eg a book, a person, a film, a country or a football team.

Table 8.4: *illustration of an entity and attributes*

ii) Attributes

The details about entities are called attributes. A person is an entity with attributes including age, height and nationality, among many others. When you design a database you need to think about which attributes you want to store. For example, the attributes of a film could include title, duration, certificate, rating, genre, cast, director and year of creation.

In a database of hotels, an individual hotel is the entity, and the attributes could be ranking, awards, location, photos, ratings and ID number.

8.6.6 Database structure

A database is organised using a set of key components. These include:

- **entities** - each recorded item
- **attributes** - details about the entity
- **field** - columns used to capture attributes
- **record** - one row of details about an entity
- **table** - a set of fields and records
- **primary key** - unique number for an entity

This is an example table of a flat-file database. The entities are films and the attributes are details about the films:

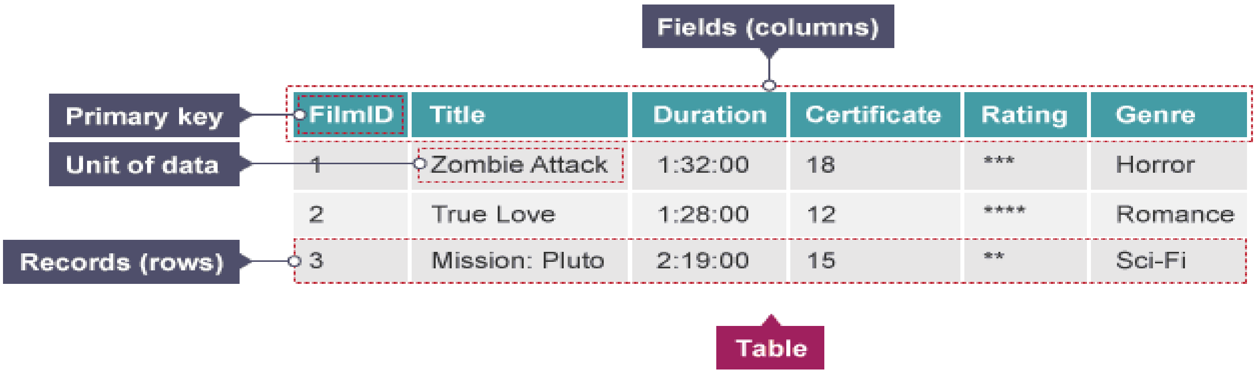


Figure 8.5: *Flat-file database illustrating the **Fields**, **primary key**, **Unit of data**, **records** and **table***

i) Table: The table contains all of the fields and the records for one type of entity. A database may contain more than one table.

ii) Records: Records contain a collection of data for each entity, usually recorded as a row in the table.

iii) Fields: The column headings are called the fields. Each field contains a different attribute. For every entity, a unit of data will be entered into each field. Each column might require different data types. For example, the 'Title' column will require data entered as text and the 'Certificate' column will need data entered as numbers.

iv) Unit of data: Each individual piece of data entered is a unit of data. These units are also called data elements.

v) Primary Key: The primary key contains a unique identifier for each record. To make each record in a database unique we normally assign them a primary key. Even if a record is deleted from a database, the primary key will not be used again. The primary key can be automatically generated and will normally just be a unique number or mix of numbers and letters.

8.6.7 Data types

The actual units of data that are entered into a database give the attributes for each entity. These units of data are also called **data elements**.

When you create a database you need to set data types for each field. For example, in a film database you might need alphabetical characters for 'Titles', but numbers for 'Duration'. Fields are usually restricted to a certain data type.

i) Data typing is a way of classifying data values that have common properties. Different kinds of data values also need different amounts of memory to store them, and have different operations that can be performed upon them.

ii) Common data types

The most commonly-supported data types are:

- integers (whole numbers), for example: 4, 27, 65535
- floating point numbers (with decimal points, sometimes called real numbers, or floats), for example: 4.2, 27.4, 56.8
- characters, for example: a, F, 3, $, £, #
- character strings (ordered sequences of characters), for example: abc, def456, 3erf78!@
- Boolean values, for example: 'True' or 'False'

iii) SQL data typing

In the database programming language SQL, the data types can be set as follows:

Data type	Explanation
char(size)	Fixed-length character string. Size is specified in brackets.
varchar(size)	Variable-length character string. Maximum size is specified in brackets.
integer(size)	Number value with a maximum number of column digits specified in brackets.
date	Date value
float(size,d)	Number value with a maximum number of digits of 'size' total, with a maximum number of 'd' digits to the right of the decimal.
Boolean	True/False values.

Table 8.6: *data types and their explanations*

8.6.8 Data Structure.

The structure of the database is also called the **schema** or **dictionary**. When you design a database you need to create a schema to explain what type of data is being stored. There are key areas to consider about the data:

- **data type** - the data types are used in each field
- **field size** - the maximum or minimum size of entry
- **validation** - the rules for accepting data
- **key field** - the field which is the primary key

Before creating a database, you need to think about its purpose. What is the database going to be used for? What searches might be performed on it?

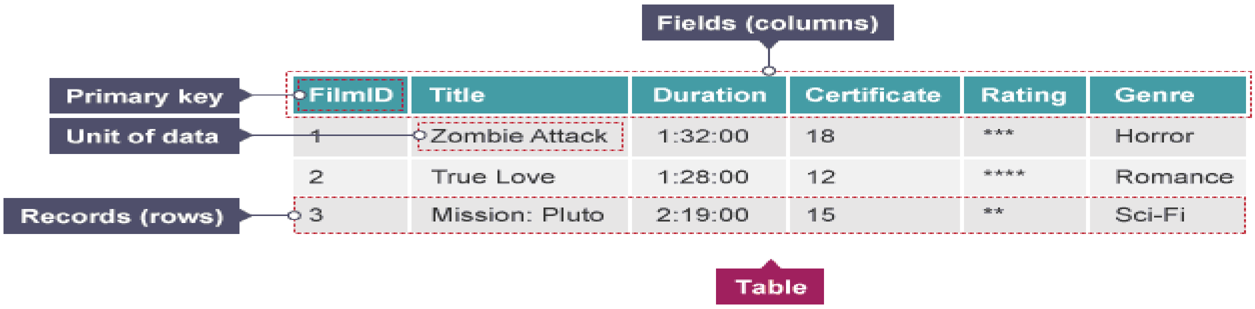


Figure 8.5: *Flat-file database illustrating the **Fields**, **primary key**, **Unit of data**, **records** and **table***

In this film table example, the structure of the table would be:

Field name	Data type	Size in bytes	Primary key?
Film ID	Integer	2	Yes
Title	Text	20	No
Certificate	Integer	2	No
Genre	Text	20	No

Table 8.7: *Data structure of the flat file database.*

Example: If you were creating a database about people, what data type would you use to capture someone's age?

Answer: *It would be best to capture their age as a date of birth with the DATE data type, eg 15/04/1998.*

An alternative would be to capture their age as a number, like 15 or 42. However, you need to think about the integrity of the database. If you input someone's age, you will need to update

each record every time someone is a year older. This would be inefficient and time consuming. It would be much better to enter their date of birth.

8.6.9 Database integrity

The integrity of the database relates to the data being valid, accurate and consistent.

Make sure your database has been designed as you want before data is entered - it is usually difficult to change a database structure after data has been entered.

A phrase database designers commonly think about is "garbage in, garbage out". This means that if you do not design your database properly, and do not take in useful information, you will not get useful information out of it.

There are two main checks to make sure a database is accurate:

- **verification**
- **validation**

Validation checks that the correct **type** of data is entered, whereas **verification** checks that the data is actually the **data you want**.

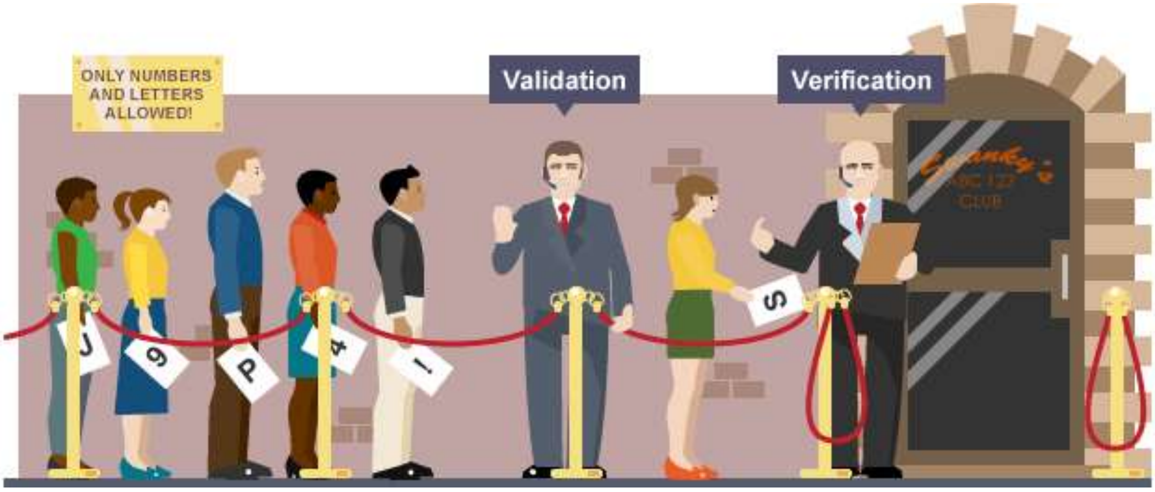


Figure 8.6: *illustration of data validation and verification*

For example, validation checks that a postcode has been entered in the correct format and verification checks that the postcode being entered is the correct postcode.

A good way of doing a verification check is by having two people entering the same data. A verification check will see if both sets are the same. If there are differences, the check will bring up an error message and ask for the data to be re-entered.

8.6.10 Relational databases

Relational databases allow data to be separated and connected across several tables. Tables are connected through primary and foreign keys to increase efficiency.

A single flat-file table is useful for recording a limited amount of data. But a large flat-file database can be inefficient as it takes up more space and memory than a relational database. It also requires new data to be added every time you enter a new record, whereas a relational database does not. Finally, ***data redundancy*** – where data is partially duplicated across records – can occur in flat-file tables, and can more easily be avoided in relational databases.

Therefore, if you have a large set of data about many different entities, it is more efficient to create separate tables and connect them with relationships.

Relational databases allow data to be stored in a clear, organised manner across multiple tables. Links, known as **relationships**, are formed to allow the data to be shared across the tables.

For instance, a retail company might have different tables for the following information:

- customer details
- customer orders
- product details
- stock levels
- stock locations
- staff details

Product details could be complicated, eg if the company sold books there may be several categories within books, including author name, title, genre, physical size and many other details. Storing all this information in one flat-file table would create a very large table

8.6.10.1 Normalisation

Normalisation is the process of analysing how to make databases more efficient by using separate tables to reduce **redundant data**. When a database is normalised, data is broken down into smaller tables and relationships are used to link them

8.6.10.2 Connecting entities

The main characteristics of a relational database are:

- it is built from a set of unique tables (also called relations)
- a table contains data about just one entity
- tables must have a primary key
- tables are linked by primary and foreign keys

When working with relational databases, users need to try to keep **information about different entities in separate tables**. Each entity has a primary key to provide a unique reference to an entity, which means that an entity can be referenced in another table without having to call up all the details about that entity.

8.6.10.3 Entity relationship diagrams

Entities can relate to each other in three different ways: **one to one, one to many and many to many.**

You can represent these relationships using an entity relationship diagram (ERD).

a) One to one

Figure 8.7: *One to one ERD*

For example, **one** person has **one** address.

b) One to many

Figure 8.8: *One to many ERD*

For example, **one** cinema has **many** customers.

c) Many to many

Figure 8.9: *Many to many ERD*

For example, **many** subjects can be taken by **many** students.

8.6.10.4 Relationship example

The following example shows how tables can be connected using primary and foreign keys. The tables are in a database for an online shop. There are three tables:

- **customer**
- **product**
- **orders**

Each table has a primary key field and each record has a primary key with a unique number.

a) Customer table

The customer table gives customers a unique Customer ID (the primary key for this table) and shows customer details, ie name, address and phone number.

Customer ID	First name	Surname	Address	Phone number
02942	Rebecca	Johnson	49 Drew Road	029 381834
02943	Mushtaq	Aqbar	28 Lyttleton Lane	028 282738

Table 8.9: *Customer table with detailed customer information*

b) Product Table:

The product table gives details about the products. The Product ID is the primary key for this table.

Product ID	Product type	Colour	Size	Cost(FCFA)
284758	Jeans	Blue	28	13,142
384957	Shoes	Brown	6	11,388
483927	Jumper	Red	M	26,292
489320	Shirt	Blue	M	27,799
839258	Socks	White	6	8,767

Table 8.10: *Product table with details about some products*

c) Orders table:

In the orders table, each order has a unique Order number (the primary key for this table). The table also includes customer ID (the primary key of the customer table) and product ID (the primary key of the product table) as **foreign keys**, but does not need to include all details about customers and products as these are stored in the Customer and Product tables.

Order number	Customer ID	Product ID	Quantity	Total cost(FCFA)
59876	02942	284758	2	26,284
59877	02942	384957	3	34,165
59878	02942	483927	1	26,293
59879	02943	489320	3	87,645

Order number	Customer ID	Product ID	Quantity	Total cost(FCFA)
59880	02943	839258	2	17,520

Table 8.11: *Orders table with normalization*

8.6.10.5 Normalization

If the Orders table did not use Customer ID and Product ID fields, it would need to include additional fields from the Customer table and the Product table – an extra eight fields. It would also need to repeat the same customer details for each order. The Orders table would be much larger, use more data, and be repetitive.

This table shows how the orders table would look *without* normalisation:

Order number	First name	Surname	Address	Phone number	Product type	Colour	Size	Cost (FCFA)	Quantity	Total cost (FCFA)
59876	Rebecca	Johnson	49 Drew Road	029 381834	Jeans	Blue	28	13,142	2	26,284
59877	Rebecca	Johnson	49 Drew Road	029 381834	Shoes	Brown	6	11,388	3	34,165
59878	Rebecca	Johnson	49 Drew Road	029 381834	Jumper	Red	M	26,292	1	26,284
59879	Mushtaq	Aqbar	28 Lyttleton Lane	028 282738	Shirt	Blue	M	29,799	3	87,645
59880	Mushtaq	Aqbar	28 Lyttleton Lane	028 282738	Socks	White	6	8,767	2	17,534

Table 8.12: *Orders table without normalization*

Normalizing the data creates a selection of simpler tables which takes up less data.

8.6.11 Database tools

Databases store data, but you also need to be able to search and filter the data to find and present results.

Some of the tools you use when working with databases include:

- **forms**
- **queries**
- **reports**
- **modules**

a) Forms

Forms are used to enter data into a database. This is only required if the database needs a user to enter data. The form should make it clear what data should be inputted, eg if you are selling something, you would fill in a form which prompts you for information like product name, brand and size.

Some databases are filled with data without direct user interaction, eg records of sensor readings from a weather station.

b) Queries

Queries are used to search and filter a database. For example, when shopping online you select the options that you need and run a search. You can also filter and sort results, eg you may want the cheapest item first, or the item with the least time left at auction. All these actions are searching, sorting and filtering – and they are all queries.

c) Reports

Reports are used to export data and present it in a way that is easy to read. For example, your address book database is full of details such as addresses, emails, dates of birth, but you might want to run a report to present just names and phone numbers.

e) Modules

Database software and languages contain **modules** - pre-written programs. However, when making a database you might think of actions you want to do that do not have a specific module. In this case, you can edit modules in the programming language and your own procedures.

8.6.12 Queries and SQL

When working with databases we need to be able to use query languages such as SQL. This requires basic use of Boolean operators.

8.6.12.1 Queries

Databases allow us to store and filter data to find specific information. A database can be queried using a variety of methods, although this depends on the software you are using.

Databases can use query languages or **graphical methods** to interrogate the data.

8.6.12.2 Query language

Query language is a *written language used only to write specific queries*. This is a powerful tool as the user can define precisely what is required in a database. SQL is a popular query language used with many databases.

8.6.12.3 Query by example (QBE)

QBE allows the user to *create queries based on a template*, usually a set of filters presented in a graphical form. If you are using database software it might have an option to connect blocks and set the filters you want. The system presents a blank record and lets you specify the fields and values that define the query.

Database management software like MySQL, **Microsoft Access** and **Oracle** have front-end graphical interfaces which make it easier to run QBE queries.

8.6.12.4 Boolean operators

In a database we often need to filter the data to group certain results. *Boolean operators* are used to filter databases using **AND**, **OR** or **NOT**. They can search multiple fields at the same time to help us retrieve the data that we need. They are used because they provide results which are 'true' or 'false'. Search engines also make use of Boolean operators to filter results.

a) AND; AND is used to search records that contain one piece of information AND another.

For example, a database of clothing could be full of different types of items. Each item of clothing has lots of attributes including price, brand, colour, quantity and material.

You might want to search for brown shoes. A query for the words *brown AND shoes* would return results that contain the words brown and shoes.

In general, search engines treat the query *brown shoes* as *brown AND shoes,* which means that all results will contain both words, eg*brown trousers and red shoes for sale*.

b) OR: OR is used to search for records that contain EITHER one piece of information OR another.It is used to request an alternative, for example *black shoes OR white shoes*. This would present results for **any shoes that were black or white**.

Most search engines use the OR function best if the search statements are defined by speech marks, eg*"brown shoes" OR "black jeans"* would show pages which either contain brown shoes or black jeans.

c) NOT: NOT is used to exclude results.The query *shoes NOT brown* will return results that *contain the word shoes but NOT the word brown*.

Some search engines use a minus sign in front of the word, instead of NOT, eg-*brown*. In these cases, if you run a search for *Doctor Who Capaldi* this will include all results with all of

the words *Doctor*, *Who* and *Capaldi*, but the search phrase *Doctor Who –Capaldi* return the same results, excluding all results for *Capaldi*.

8.6.12.5 Arithmetic operators

A query can also be performed using *arithmetic operators*. These help to make specific searches related to numerical data.

a) Functions of arithmetic operators

This table shows some arithmetic operators and their functions:

Operator	Meaning
=	Equals
<	Less than
<=	Less than or equal to
>	Greater than
>=	Greater than or equal to
<>	Not equal to

Table 8.13: *Table with some arithmetic operators*

b) Using Arithmetic Operators

The table below shows some BBC TV programme listings:

ID	Title	Genre	Duration (mins)	Channel
01	EastEnders	Drama	30	BBC1
02	Dragons' Den	Entertainment	60	BBC2
03	The Voice	Entertainment	75	BBC3
04	Blue Peter	Children's	25	CBBC
05	Wild Brazil	Nature	60	BBC4
06	Match of the Day	Sport	80	BBC1
07	Dick and Dom	Comedy	10	CBBC

Table 8.14: *BBC TV programme listings*

Queries are useful for searching for specific conditions. You might want to find entertainment programmes on BBC3. A query for these conditions would look like this:

```
SELECT * FROM Programmes
WHERE Genre='Entertainment'
AND Channel='BBC3';
```

This would return the programme 'The Voice'.

You may want a programme that is less than 20 minutes long or is a nature programme. A query for these conditions would look like this:

```
SELECT * FROM Programme
WHERE Duration<20
OR Genre='Nature';
```

This would return the programmes "Dick and Dom" and "Wild Brazil"

8.6.12.6 SQL

SQL is a programming language used to search and query databases. SQL gives you the ability to customise your queries. It is often used within database programs.

a) Uses of SQL

SQlite and MySQL are popular open source database applications. If you use a blog site, it is very likely that MySQL has been running behind the scenes to store entries as records and allow you to add, edit and delete blog posts.
SQL databases can be used to create lots of applications for use on the internet. They are often used by large companies because they allow the data tables to be stored on secure servers. A SQL server simply stores the data for a database - it does not provide front-end features. Therefore, the user does not need to know that a database is working behind a website.

A SQL database is manipulated using SQL code. SQL has also been designed so that lots of users can access it at the same time - it has a high capacity for storage space.

b) Syntax

SQL is an important programming language and understanding how to use it is a very important skill.
The statements used in SQL code are quite similar to the words that you would normally use to find information. Because the statements are quite similar, it makes it easier to remember SQL syntax.
For example: "**SELECT** these fields **FROM** this table **WHERE** this is happening".

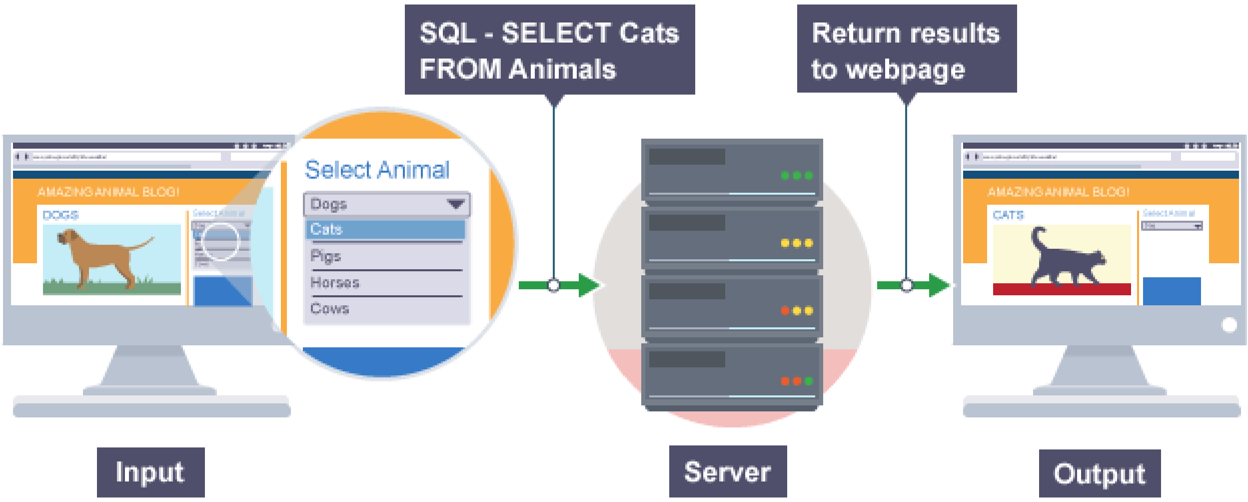


Figure 8.10: *illustration of SQL*

(i) Working with SQL: Step 1

The following examples show how to use SQL for basic database functions. We will work through a series of steps to create this table, showing a record for a collection of BBC programmes:

ID	Title	Genre	Duration
01	EastEnders	Drama	30
02	Newsnight	Current affairs	50
03	The Voice	Entertainment	75
04	Blue Peter	Children's	25
05	Wild Brazil	Nature	60
06	Sherlock	Drama	90
07	Top Gear	Entertainment	60

Table 8.15: record for a collection of BBC programmes

Creating a table

A table can be created in SQL code using the following template:

```
CREATE TABLE tablename
(column1 datatype,
column2 datatype,
column3 datatype);
```

The creator of the table has to decide:

- the name of the table - "tablename"
- the title of each field - "column"
- the data type that is required for each field - "datatype" (eg character strings - 'varchar' or numbers - 'int')

In the TV programmes example the table can be created using the following SQL code:

```
CREATE TABLE Programmes
(ID int(2),
```

```
Title varchar(20),
Genre varchar(20),
Duration int(3));
```

The data type *varchar* indicates that only character strings are allowed, and the number in brackets indicates the maximum number of digits for each data type.

(ii) Working with SQL: Step 2

You can query this table using SQL code.

ID	Title	Genre	Duration
01	EastEnders	Drama	30
02	Newsnight	Current affairs	50
03	The Voice	Entertainment	75
04	Blue Peter	Children's	25
05	Wild Brazil	Nature	60
06	Sherlock	Drama	90

You may decide that you wish to sort the programmes by duration. The SQL code needed would look like the example below:

```
SELECT Programmes.Duration, Programmes.Title
FROM Programmes
ORDER BY Programmes.Duration;
```

- the **SELECT** statement states which fields to look at - the Title and Duration fields
- the **FROM** statement states which table to look at - Programmes
- the **ORDER BY** statement sorts the Duration field in ascending order by default

This table shows the results from this query:

Duration	Title
25	Blue Peter
30	EastEnders
50	Newsnight
60	Wild Brazil
75	The Voice
90	Sherlock

Table 8.16: results from query made on table 8.15

(iii) Working with SQL: Step 3

The SQL **WHERE** statement is used to isolate one record or several records with similar attributes.

ID	Title	Genre	Duration
01	EastEnders	Drama	30
02	Newsnight	Current affairs	50
03	The Voice	Entertainment	75

04	Blue Peter	Children's	25
05	Wild Brazil	Nature	60
06	Sherlock	Drama	90

Table 8.15: record for a collection of BBC programmes

The following code searches the Title field of the table to find the words 'The Voice'.

```
SELECT Programmes.ID, Programmes.Title, Programmes.Genre,
Programmes.Duration
FROM Programmes
WHERE ((Programmes.Title)="The Voice");
```

The **WHERE** statement specifies which text to look for

This table shows the results from this query:

ID	Title	Genre	Duration
03	The Voice	Entertainment	75

Alternatively, you could find all of the programmes which are less than 30 minutes long using this code:

```
...
WHERE ((Programmes.Duration)<30);
```

This table shows the results from this query:

ID	Title	Genre	Duration
04	Blue Peter	Children's	25

(a) Wildcards

The wildcards uses the * symbol, and is used in place of any number of unknown characters. For example, the following code searches for all programmes with the letter **i** in the title:

```
...
WHERE ((Programmes.Title) LIKE "*i*");
```

This table shows the results from this query:

ID	Title	Genre	Duration
02	Newsnight	Current Affairs	50
03	The Voice	Entertainment	75
05	Wild Brazil	Nature	60

(b) Adding and editing data with SQL

SQL can also be used to add and edit the data stored on an SQL server.

(i) Adding records

To add new data you use the function **INSERT INTO**.

If a new BBC programme is created and you want to add it to the database, then you would need to use the INSERT INTO function followed by the VALUES separated by a comma:

```
INSERT INTO Programmes
VALUES (07, "Top Gas", "Entertainment", 60);
```

NB: *quotes surround string entries. Numbers do not have quotes*

After inputting this code the table would look like this:

ID	Title	Genre	Duration
01	EastEnders	Drama	30
02	Newsnight	Current affairs	50
03	The Voice	Entertainment	75
04	Blue Peter	Children's	25
05	Wild Brazil	Nature	60
06	Sherlock	Drama	90
07	Top Gas	Entertainment	60

(ii) Editing records

SQL also has the **UPDATE** function for editing data.

In this example, there was an error with the previous entry and you need to change the name from "Top Gas" to "Top Gear". You need to:

1. Identify the table to be updated using **UPDATE** - `UPDATE Programmes`
2. Identify what the field needs to be changed to using **SET** - `SET Programmes.Title = "Top Gear"`
3. Identify which record needs to be updated using **WHERE** - `WHERE Programmes.ID = 07`

In full, this is:

```
UPDATE Programmes
SET Programmes.Title = "Top Gear"
WHERE Programmes.ID = 07;
```

This will go to the *Programmes* table, find the programme with an **ID** of 07 and change the *Title* field to *Top Gear*. The amended table will look like this:

ID	Title	Genre	Duration
01	EastEnders	Drama	30
02	Newsnight	Current affairs	50
03	The Voice	Entertainment	75
04	Blue Peter	Children's	25
05	Wild Brazil	Nature	60
06	Sherlock	Drama	90
07	Top Gear	Entertainment	60

8.6.13 Database Management System (DBMS)

A database management system (DBMS) is a tool to store, edit and organize data in a database. It provides several key features:

i. stores data in one central location
ii. allows data to be shared by many users
iii. provides user interface to work with the data
iv. Creates backups.
v. controls who can access and edit the data

8.6.13.1 Benefits of a DBMS

The benefits of a DBMS include:

i. **integrity** - the structure of the database can change, but the applications using the data do not need to be changed
ii. **efficiency** - avoids data duplication and inconsistency, and less storage space is taken up because data is shared
iii. **consistency** - data is the same, regardless of who is viewing it
iv. **backups** - it is easy to back up data from one location
v. **security** - the data is in a secure central place and different access rights can be assigned to different people
vi. **customisation** - applications can be customised to suit the needs of the user

An important part of a DBMS is separating applications from the data. When people use the applications they call on the data they need to work on. They do not need to use all the data every time they use the database.

A database in a DBMS could be viewed by lots of different people with different responsibilities.

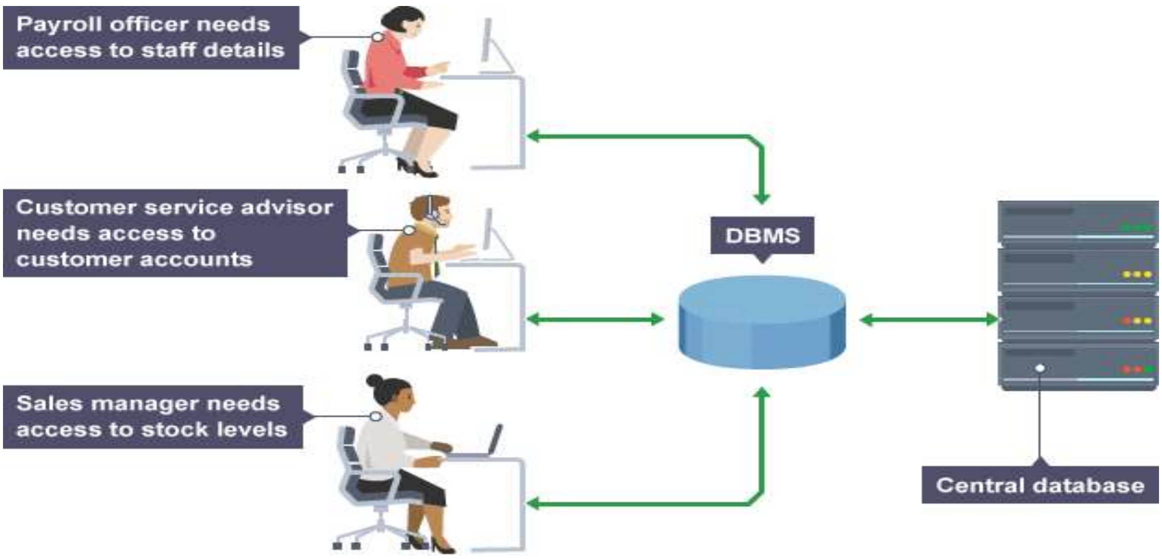


Figure 8.11: *Applications of DBMS in an organization*

For example, within a company there are different departments, as well as customers, who each need to see different kinds of data. Each employee in the company will have different levels of access to the database with their own customized front-end application.

8.6.13.2 DBMS example

The following example shows how a DBMS is used for a ticketing website.A ticketing company allows customers to buy tickets online or through a booking office. *How can they use a DBMS to keep a record of sales?*

(i) Option 1

They could have two separate databases: one for online sales and one for booking office sales. But they would both need to keep records of tickets and customers.

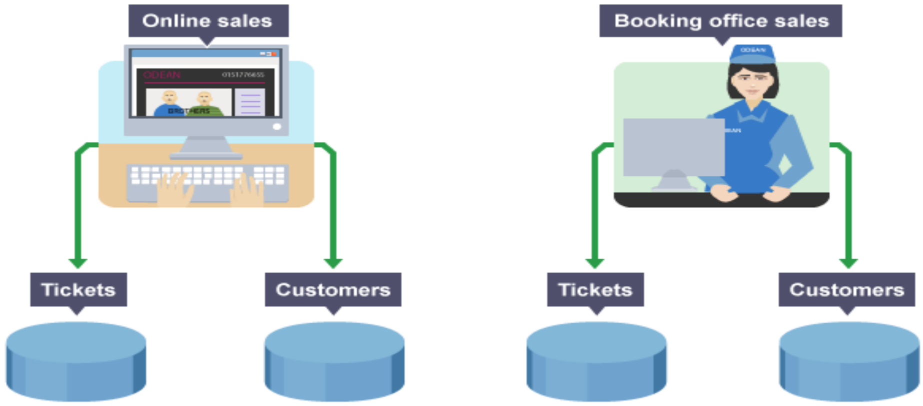


Figure 8.12(a): *Application of DBMS for a ticketing website*

This type of system would **not** work effectively for a number of reasons:

- How would you know which seats had been sold in each system?
- Would you divide the seats up before offering them to the public?
- What would happen if online tickets sold out before booking office tickets did?

You could lose customers due to errors or double-selling the same tickets.

(ii) Option 2

A DBMS could create a system where the ticket and customer data could be held in one database. There could be an application for the booking office and the online sales which would allow both of them to access the database. Data could then be shared between online sales and booking office sales and the data would not be duplicated.

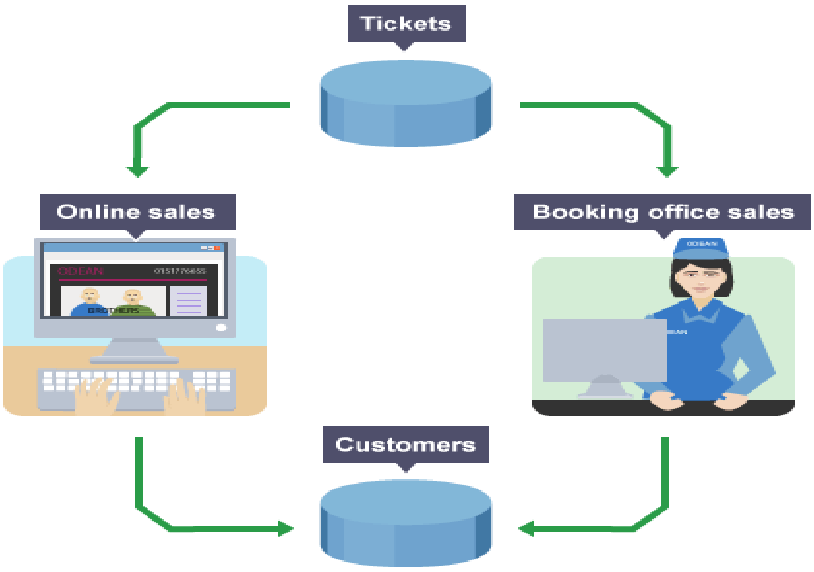


Figure 8.12(b): *Application of DBMS for a ticketing website*

This type of system would work much more effectively.

8.6.13.3 Database views

One of the benefits of using a DBMS is that it allows different views of a database. This is useful as different users have different requirements.

The *external* view is the view of the data used by customers or people who input and view data, a *conceptual* view is useful for the database designer and the *internal* view is used for the detailed programming of the database.

A customer does not need to see as much information as the administrator or programmer of the database.

(i) External view

The external view is the view that the **customer** would see. If you made an app which used a database to collect data, you might want to give users a simple form to add new information to the app.

For example, if there was an app which people used for bird watching it could simply have a form which includes:

- bird type
- time of year
- date
- location

The app user does not need to see the whole database to add their information. It is possible for a database user to work with a database and never actually see the tables that are used to store the data.

(ii) Conceptual view

This view shows how the tables are all connected. It could be used to plan changes to the database. This view can also be presented using SQL.

(iii) Internal view

This is the view that shows the detail of how the computer sees the database. It might include details of how files are structured and stored and include hex or binary references to bytes of data. It is complex and would only be used by the database programmer.

8.6.13.4 Access levels

One of the challenges of creating a DBMS is managing who can access and change the data. If anyone can edit the database, data could easily become corrupted. It is better to have different levels of access for the database to maintain the integrity of the database.

For example, with personalized websites that require a login, you will only see data that is relevant to you. However, if an employee of that website company accesses the DBMS, they will be able to view all customer accounts. The customer details come from the same database, but different levels of access are given to different users.

A DBMS developer would think about the needs of the user and develop a bespoke system to suit the needs of the company.

With an online music library, there would be different access levels for different users. For example:

i. **customers** should only see their own music
ii. **administrators** would have access to upload and amend the entire library
iii. **account managers** would be able to see the financial details for the customers when required

Websites use database applications to customise the user interface to suit the needs of each user. The owner of a blog will be able to delete, add and edit comments, whereas the reader would simply be able to view the blog posts.

With social networks you customise who can access your data:

i. you might only allow 'friends' to see your data
ii. you might allow 'friends' and 'friends of friends' to see your data
iii. you might have a public account that is visible to everyone

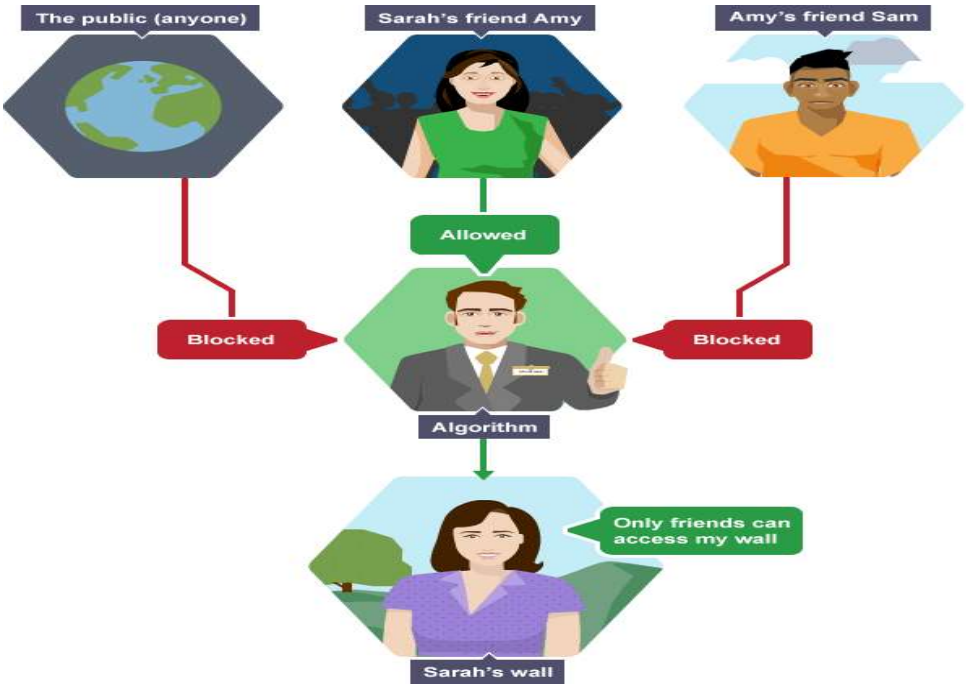


Figure 8.13: *Illustration of access levels on a social network*

The data that appears on Sarah's profile on a social network is stored in a database. Sarah has set up access rights to her data. Sarah only allows users known as her 'friends' to view her profile. This means that the public and 'friends of friends' cannot see her data.

(a) Concurrency and ACID

Concurrency is when many users are interacting with a system or many changes are being made at once.

A DBMS needs to use concurrency. However, the database needs to prevent two people using the data in a conflicting way.

For example, two people visiting an online shop might try to buy the final product in store. The database would need to make sure only one person could buy it.

(b) ACID rules

A change in a database is called a **transaction**. Changes to databases must conform to ACID rules:

i. **Atomicity** - the transaction must be completed fully. If it is not completed fully it will not be recorded.
ii. **Consistency** - any change must not break the database. It must be consistent with how it was before the change.
iii. **Isolation** - a transaction must be isolated and not interfere with another transaction.
iv. **Durability** - a transaction must remain in the database.

Drill Questions:

1. What are numbers, pictures or words without context known as?
a) data (b) Information (c) Data

2. What are a collection of words, numbers or pictures with meaning known as?
a) data (b) Information (c) Data

3. What does putting data into a framework or structure provide?
a) More Information (b) Rules (c) Context

4. What is the ability to understand information and to then form judgements, opinions, make predictions and decisions based on that understanding also known as?
a) data (b) Information (c) Data

5. Which of the following is correct?
(a) Data leads to information, and Information leads to knowledge.
(b) Knowledge Leads to information, and information leads to knowledge.
(c) Data leads to knowledge and knowledge leads to information.

6. The statement 'it was raining outside' is which of these?
a) data (b) Information (c) Data

7. Which of the following is NOT an example of data?
a) 301062 (b) Blue (d) 32, Primrose Hill

8. 'The petrol gauge was showing almost empty. I decided to look out for the next petrol station'. What is the statement 'I decided to look out for the next petrol station' an example of?
a) data (b) Information (c) Data
9. What is a computer program that makes decisions based on the large quantity of information it stores known as?
a) A Specialist System (b) An Expert System (c) A decision Making System

10. Which is the most suitable program for storing large quantities of data?
a) Database (b) Presentation (c) Animation

Exercise 11:

i) Dates are read into a computer in the following format: DDMMMYY e.g. 15DEC92. The following dates were rejected by a validation program: 3JAN71, AUG2166, 31SEP72. State the validation check used to discover each error.
ii) Calculate the check digits to complete the ISBNs: 019276150-**1** and 995640216-**8**.
iii) Set up a spreadsheet which will calculate and check the validity of Modulo-11 weighted check digits for any given ISBN.

12. (a) what is an Information system?
(b) Explain the term “procedure” as a component of an information system.
(c) Interms of Information system describe GIS, MIS and LIS. (**Q2ii, CGCE 2016**)

13. (a) Describe a DBMS.
(b) What is relational database?
(c) Give two disadvantages of a flat file in databases.
(d) with respect to relational database, discuss with the aid of diagrams, the meaning of the following:-
(i) One-to-one relationship
(ii) One-to-many relationship
(iii) Many-to-many relationship (**Q1ii,Q5i, CGCE 2015**)

Chapter 9: DATA COMMUNICATIONS

9.1 Introduction

Data communication refers to the exchange of data between two devices via some form of communication channel. In data communication the following basic terms are frequently used:

1. **Data**: a collection of facts in raw form that becomes information after processing.
2. **Signal**: an electric or electromagnetic encoding of data.
3. **Signaling**: propagation of signals across a communication channel.
4. **Transmission**: sending of data from one place to another by means of signals.

There are five basic components in a communication system.

i. **Message**: the message is the information to be communicated;
ii. **Sender**: the sender is the device that sends the data message;
iii. **Receiver**: the receiver is the device that receives the message;
iv. **Transmission Medium**: the transmission medium is the physical path by which a message travels from sender to receiver.
v. **Protocol** - It is a set of rules that governs the data communications. It represents an agreement between the communicating devices. Without a protocol, two devices may be connected but not communicating.

Example: John calls Peter on phone.
The data source is John, the transmitter is John's phone, the communication channel is the telephone cable or microwave, the receiver is Peter's phone and the destination is Peter.

9.2 Analog and Digital Signals

Data is transmitted from one point to another by means of electrical signals that may be in analogue or digital form.

9.2.1 Analogue Signals

An analogue signal is one in which information is represented as a continuous variation of some physical property or quantity. Analogue signals are continuous waves that carry information by varying the frequency or amplitude of the wave.

i. When the amplitude of the signal is varied the technique is called amplitude modulation (AM)
ii. When the frequency of the signals is varied, the technique is called frequency modulation (FM).

Human speech is an example of an analog signal. Telephone lines use analog signals because they were originally designed for speech.

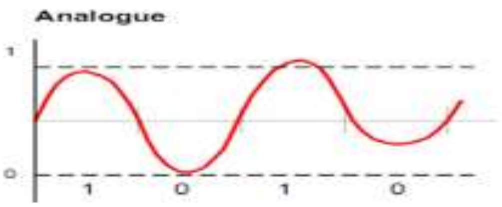


Figure 9.1: *Analogue signal*

9.2.2 Digital Signals

A digital signal is one in which information is represented as a sequence of binary values 0 and 1.These two values represent two conditions, on or off, corresponding to two known levels of voltage or current.

Digital signals do not continuously vary as analogue signals. Signals are transmitted within the computer as digital signals. Systems that use digital technology are known as baseband systems.

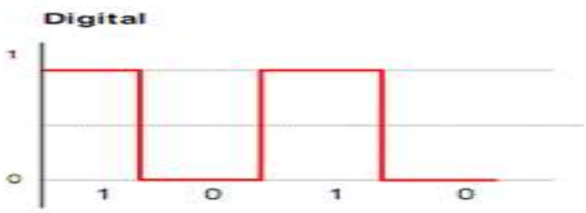


Figure 9.2: *Digital Signal*

9.3 Broadband and Baseband Transmissions

9.3.1 Baseband System

A baseband system is a single-channel system that supports a single transmission at any given time. In a baseband system, data is sent as a digital signal through the media as a single channel that uses the entire bandwidth of the media. Baseband communication is bi-directional, which means that the same channel can be used to send and receive signals. In Baseband, frequency-division multiplexing is not possible.

9.3.2 Broadband System

A broadband system is a system that supports multiple transmissions via multiple frequency channels. In a broadband system, data is sent in the form of an analog signal where each transmission is assigned a portion of the bandwidth. Broadband communication is unidirectional, so in order to send and receive, two pathways are needed. This can be accomplished either by assigning a frequency for sending and assigning a frequency for receiving along the same cable or by using two cables, one for sending and one for receiving.

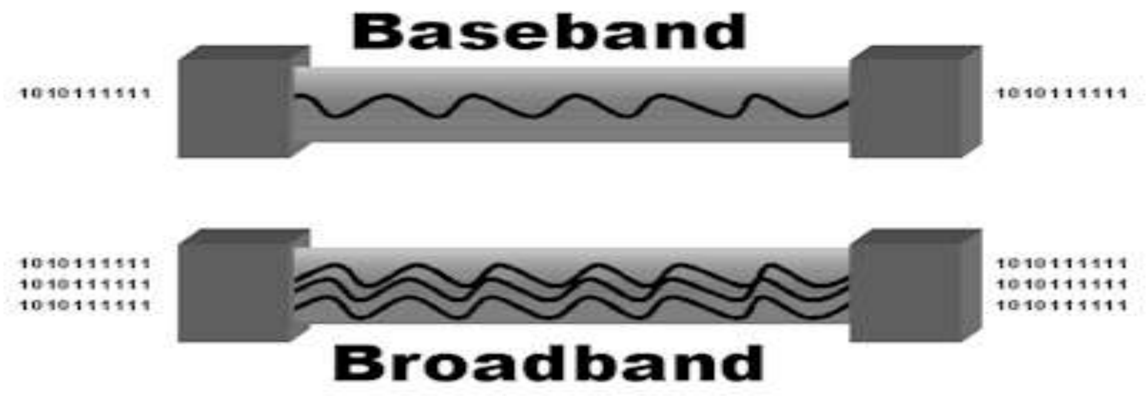


Figure 9.3: *Baseband and Broadband Transmission*

9.4 Transmission Modes and Techniques

Transmission modes simply refer to the direction of flow of information between two communicating devices. It could be simplex, half duplex or full duplex.

9.4.1 Simplex

In simplex mode, signals are transmitted in only one direction. The flow of information is unidirectional from transmitter to receiver always. Examples are television broadcasting, computer to the printer connection and CPU to monitor communication.

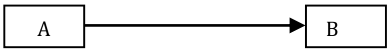


Figure 9.4: *Simplex transmission mode*

9.4.2 Half Duplex

In half duplex mode, signals can be transmitted in both directions but only one way at a time. The flow of information is bidirectional but information can only be sent if it is not being received. It is suitable for data transmission between a computer and dumb terminals. An example is the police radio (walkie-talkie).

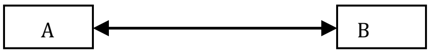


Figure 9.5: *Half Duplex transmission mode*

9.4.3 Full Duplex

In full duplex mode, signals can be transmitted in both directions simultaneously. The communicating devices can transmit at the same time. The flow of information is bidirectional. It is suitable for interactive systems. An example is the telephone.

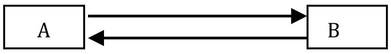


Figure 9.6: *Full Duplex transmission mode*

9.4.4 Parallel Transmission

Parallel transmission is the method of transferring several bits at the same time over separate channels. For example, eight separate channels will be required if a block of eight bits is to be transmitted in parallel. Parallel transmission is fast but it is suited only for short distances as cabling for long distances will be expensive. It is mainly used for connections within the computer and for connecting the computer to the printer (parallel printer port and cable). Most printers are within 6 meters or 20 feet of the transmitting computer and the slight cost for extra wires is offset by the added speed gained through parallel transmission of data.

9.4.5 Serial Transmissions

Serial transmission is the method of transferring data one bit at a time through the same channel. If a block of 8 bits is to be transmitted in series, the bits will be transmitted one after the other on the same channel. Serial transmission can be *asynchronous* or *synchronous*.

i) Asynchronous Serial Transmission

Asynchronous transmission describes the process where transmitted data is encoded with start and stop bits, specifying respectively the beginning and end of each character. Data is sent character by character with each character preceded by a start bit and a stop bit is added to the end. Other control bits like the parity bit are added to the group before the stop bit and small gabs are inserted to distinguish each group.

ii) Synchronous Serial Transmission

Synchronous transmission describes a continuous and consistent timed transfer of data blocks. Data is sent as one long bit stream or block of data without start or stop bits and with no gabs. Upon reception, the receiver counts the bits and reconstructs bytes. It is essential that the same timing is maintained by both sender and receiver as there are no start and stop bits and no gaps. Another channel is therefore used to transfer timing signals to keep the both parties synchronized. Accuracy is dependent on the receiver keeping an accurate count of the bits as they arrive.

Characteristic features of a transmission system are the *direction of the data flow* and the *data throughput*, or the *maximum possible data rate*.

a) Direction of data flow: Transmission systems differ as to the direction in which the data flow and when messages can be transmitted. Basically, there are three different ways of communication: *Simplex, Half duplex and Full Duplex*.

b) Data transmission speed: An essential criterion for determining the capacity of communication lines is the data rate, i.e. the speed at which the data can be transmitted. The data rate is characterized by the number of bits transmitted each second, measured in *bps*, *bits per second*. As data rates are extremely high nowadays, such units as *kilobit per second; kbit/s* and *megabit per second; Mbit/s* are not unusual.

When each bit is encoded and transmitted individually, the transmission line must be able to transmit frequencies that correspond to half of the bit transmission rate:

- bit transmission rate: 100 kbit/s
- transmission frequency: 50 kHz

When it is necessary to achieve a high data rate, even though the transmission bandwidth is limited, several bits can be grouped and encoded together.

Serial transmission is between two computers or from a computer to an external device located some distance away. Examples of serial mode transmission include connections between a computer and a modem using the *RS-232 protocol*.

Comparatively, serial transmission is slower than parallel transmission but it is suited for long distances. It is cheaper as only one transmission line is required. Synchronous transmission is faster than asynchronous transmission because fewer bits have to be transmitted; only data bits and no extra control bits. For this reason it is the choice for network communications links.

9.4.6 Multiplexing

Multiplexing, in communication, technique whereby two or more independent messages, or information-bearing signals, are carried by a single common medium, or channel. When multiplexing is performed, two or more channels are combined into a single channel, or, in a process often called **demultiplexing**, a single channel is divided into several sub-channels. Many different types of multiplexing are possible;

i) Frequency-division multiplexing, in which a single frequency channel is subdivided into two or more sub-channels, each of which can then carry a smaller range of frequencies than could the original channel. Frequency-division multiplexing is used in *television broadcasting*,

when audio and video signals share a single channel; in *stereophonic FM radio broadcasting*, when two audio signals share a single channel; and in *microwave transmission of long-distance telephone calls*, when 60 or more conversations are carried by a single microwave beam.

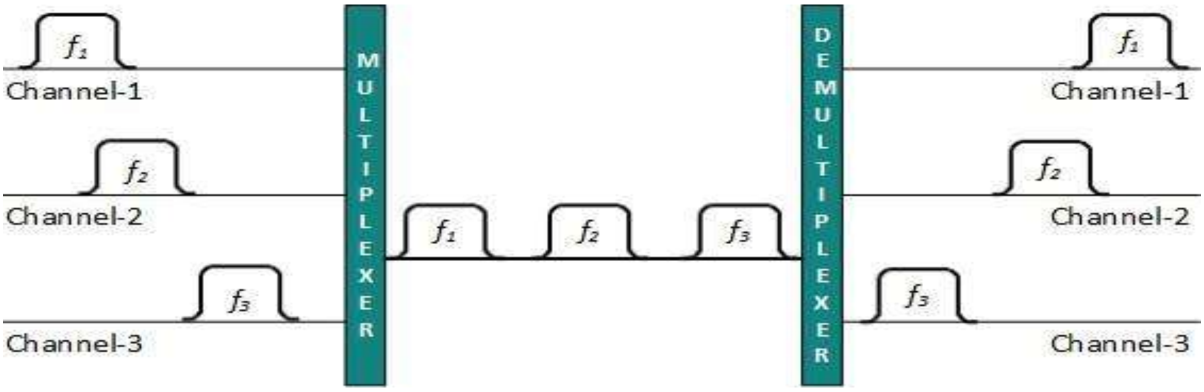


Figure 9.7: *Frequency Division Multiplexing (FDM) technique*

ii) Time-division multiplexing, in which successive small time intervals are used for the transmission of messages over a single channel. Time-division multiplexing is often used in the construction of digital computers. When information can be stored into or retrieved from the computer's memory at a much greater rate than it can be supplied or used by an external device such as a card reader, printer, or teletype terminal, several such low-speed devices can share a single multiplexed data channel.

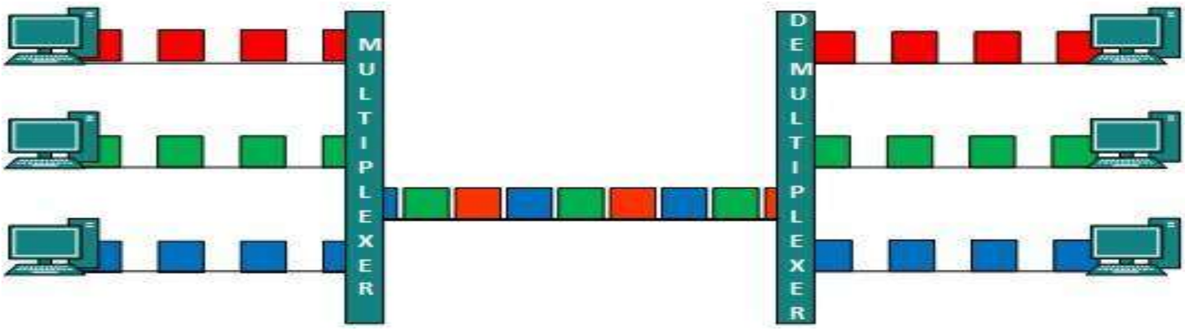


Figure 9.8: *Time Division Multiplexing (TDM) Technique*

iii) Wavelength-division Multiplexing, Wavelength division multiplexing is a technique where optical signals with different wavelengths are combined, transmitted together, and separated again. It is mostly used for *optical fiber communications* to transmit data in several (or even many) channels with slightly different wavelengths. In this way, the transmission capacities of fiber-optic links can be increased strongly, so that most efficient use is made not only of the fibers themselves but also of the active components such as fiber amplifiers. Apart from telecom, wavelength division multiplexing is also used for, e.g., *interrogating multiple*

fiber-optic sensors within a single fiber.

Figure 9.9: *Wavelength Division Multiplexing (WDM) Technique*

9.5 Communication Switching Techniques

Long distance transmission is done over a network of switched nodes. Data is routed by being switched from one node to another. Three switching techniques exist: *packet switching*, *circuit switching* and *message switching*.

9.5.1 Packet Switching

Packet switching is a switching method in which the message to be transmitted is broken into small data packets and sent over the network. Each packet contains a portion of data and some control information. The packets may take different routes to arrive their destination and they may arrive in any other. On arrival, they are put back into order and the message is reconstituted. Each packet is sent with a header address which tells what its destination is. The header address also describes the sequence for reassembly at the destination. One packet contains information on how many packets should be arriving. If a packet fails to arrive, the destination computer sends a message to the sender’s computer asking it to send the missing packet again. This method is suitable for transmission of data.

9.5.2 Circuit Switching

Circuit switching is a switching method in which a dedicated communication path in physical form between two stations within a network is established, maintained and terminated for each communication session. This channel remains open throughout the communication process and cannot be used by anyone else. It has basically three phases: circuit establishment, data transfer and circuit disconnect. The message is sent without being broken up, so it is received in the order it was sent. This method was designed for voice transmissions. Telephone networks use circuit switching for transmission of phone calls.

9.6 Transmission Media

A transmission medium is the physical pathway that connects computers and other devices on a network. Each transmission medium requires specialized network hardware that is compatible with that medium, and most networks need to use a combination of transmission media types selected based on the network's needs. There are two categories of transmission media: guided and unguided media.

9.6.1 Guided Media

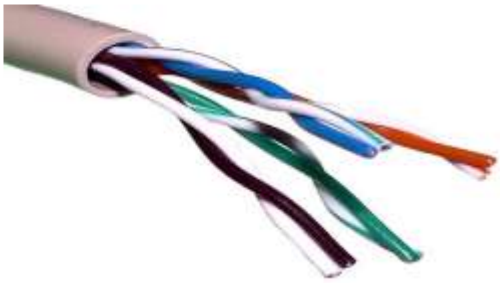
Guided media are the physical links through which signals are confined to narrow path. They are made up of an internal conductor bounded by jacket material. They are also called bounded or conducted media. Three common types of guided media are *coaxial cable*, *twisted pair cable, Power Line cable* and *fiber optical cable*.

i) Twisted Pair Cable: A twisted pair cable is made of two plastic insulated copper wires twisted together to form a single media. Out of these two wires, only one carries actual signal and another is used for ground reference. The twists between wires are helpful in reducing noise (electro-magnetic interference) and crosstalk.

There are two types of twisted pair cables:

- **Shielded Twisted Pair (STP) Cable**
- **Unshielded Twisted Pair (UTP) Cable**

STP cables comes with twisted wire pair covered in metal foil. This makes it more indifferent to noise and crosstalk.



`

Figure 9.10: *Twisted Pair Cable*

UTP cables consist of 2 or 4 pairs of twisted cable. Cable with 2 pair use *RJ-11 connector* and 4 pair cable use *RJ-45 connector*. RJ stands for registered jack. UTP has **seven** categories, each suitable for specific use;

Category 1: These are used in telephone lines and low speed data cable.
Category 2: These cables can support up to 4mbps implementation.
Category 3: These cable supports up to 16mbps and are mostly used in 10mbps.
Category 4: These are used for large distance and high speed. It can support 20mbps.
Category 5: This is the highest rating for UTP cable and can support up to 100mbps.

In computer networks, nowadays, **Cat-5, Cat-5e, and Cat-6 cables** are mostly used. UTP can be connected as straight through or crossover. A straight-through cable has identical ends. A crossover cable has different ends.

ii) Coaxial Cable: Coaxial cable has two wires of copper. The core wire lies in the center and it is made of solid conductor. The core is enclosed in an insulating sheath.The second wire is wrapped around over the sheath and that too in turn encased by insulator sheath.This all is covered by plastic cover.

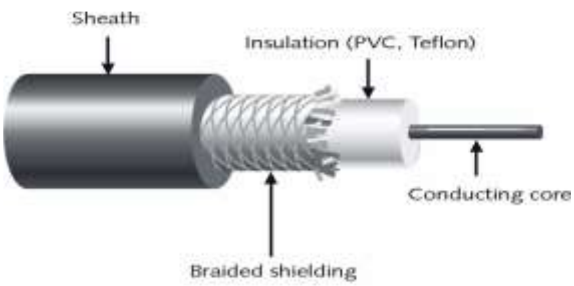


Figure 9.11: *Coaxial Cable*

Because of its structure, the coax cable is capable of carrying high frequency signals than that of twisted pair cable.The wrapped structure provides it a good shield against noise and cross talk. Coaxial cables provide high bandwidth rates of up to 450 mbps.

There are three categories of coax cables namely, **RG-59 (Cable TV)**, **RG-58 (Thin Ethernet)**, and **RG-11 (Thick Ethernet)**. RG stands for *Radio Government*.Cables are connected using *BNC (Bayone-Neill-Concelman)* connector and *BNC-T*. *BNC terminator* is used to terminate the wire at the far ends.

iii) Power Lines: Power Line communication (PLC) is Layer-1 (Physical Layer) technology which uses power cables to transmit data signals. In PLC, modulated data is sent over the cables. The receiver on the other end de-modulates and interprets the data.

Because power lines are widely deployed, PLC can make all powered devices controlled and monitored. PLC works in *half-duplex*.There are two types of PLC: **Narrow band PLC** and**Broad band PLC**

- **Narrow band PLC** provides lower data rates up to 100s of kbps, as they work at lower frequencies (3-5000 kHz).They can be spread over several kilometers.

- **Broadband PLC** provides higher data rates up to 100s of Mbps and works at higher frequencies (1.8 – 250 MHz).They cannot be as much extended as Narrowband PLC

iv) Optical Fibers: Fiber Optic works on the properties of light. When light ray hits at critical angle it tends to refracts at 90 degree. This property has been used in fiber optic. The core of fiber optic cable is made of high quality glass or plastic. From one end of it light is emitted, it travels through it and at the other end light detector detects light stream and converts it to electric data.

Fiber Optic provides the highest mode of speed. It comes in two modes, one is single mode fiber and second is multimode fiber. Single mode fiber can carry a single ray of light whereas multimode is capable of carrying multiple beams of light.

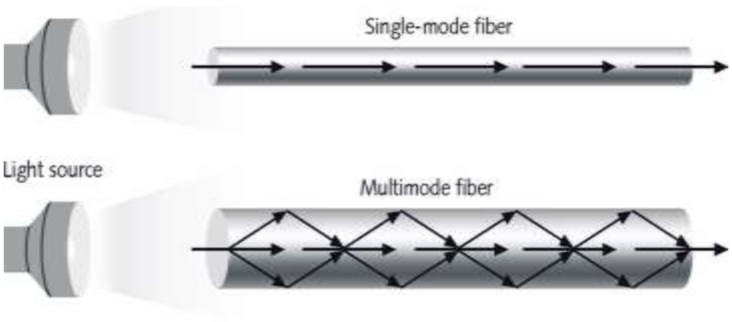


Figure 9.12: *Fibre Optic modes of transmission; Single Mode Fiber, Multimode Fiber*

Fiber Optic also comes in unidirectional and bidirectional capabilities. To connect and access fiber optic special type of connectors are used. These can be Subscriber Channel (SC),

Straight Tip (ST), or MT-RJ.The light source can be *LED (light emitting diode)* or *LD (laser diode)*

9.6.2 Unguided Media

Unguided media do not use physical means to define the path to be taken. They provide a means for transmitting electromagnetic waves but do not guide them. They are also called unbounded media. Examples of unguided media are infrared waves, radio waves and microwaves.

i) Infrared

Infrared uses transmitters/receivers (transceivers) that modulate non-coherent infrared light. Infrared signals do not penetrate walls as such transceivers must be within line-of-sight either directly or via reflection. Line of sight is a type of propagation that can transmit and receive data only where transmit and receive stations are in view of each other without any sort of an obstacle between them.

ii) Radio waves

Radio wave systems transmit signals by modulation of electromagnetic waves with frequencies below that of visible light. Radio waves carry information by systematically changing some property of the radiated waves such as amplitude (AM radio), frequency (FM radio) and phase. Radio waves are omnidirectional. This means that signals spread out in all directions and can be received by many antennas.

iii) Microwaves

Microwaves are electromagnetic radiations beyond the frequency range of radio and television. There are two types of microwave systems: *terrestrial microwave systems* and *satellite systems*.

a) Terrestrial microwave systems are land-based. Microwaves being line-of-sight and traveling in a straight line, the earth's curvature poses a problem to long distance microwave transmissions. As such, long distance transmissions require directional antennas (repeaters)to be used at intervals of 25 to 30 kilometres between the transmitting and receiving end.

b) Satellite systems use communication satellites to solve the problem posed by the earth's curvature to terrestrial microwave systems. A communication satellite is a microwave relay station placed in outer space. A microwave signal is transmitted from earth to the satellite which amplifies the signal and sends it back to earth. The earth station transmits the signal to the satellite on an *up-link,* on one frequency and the satellite repeats those signals on a *down-link* which is on another frequency.

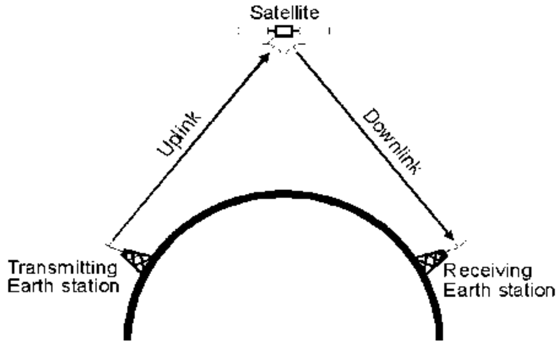


Figure 9.13: *Satellite Systems*

✓ **Advantages of microwave systems**
- No cables needed
- Multiple channels available
- Wide bandwidth

✓ **Disadvantages of microwave systems**
- Line-of-sight will be disrupted if any obstacle, such as new buildings, are in the way
- Signal absorption by the atmosphere. Microwaves suffer from attenuation due to atmospheric conditions.
- Towers are expensive to build

9.7 Connection Vs Connectionless Transmission:

9.7.1 Connection-oriented:

Requires a session connection (analogous to a phone call) be established before any data can be sent. This method is often called a "*reliable*" network service. It can guarantee that data will arrive in the same order. Connection-oriented services set up virtual links between end systems through a network, as shown in*Figure 9.14*. Note that the packet on the left is assigned the virtual circuit number 01. As it moves through the network, routers quickly send it through virtual circuit 01.

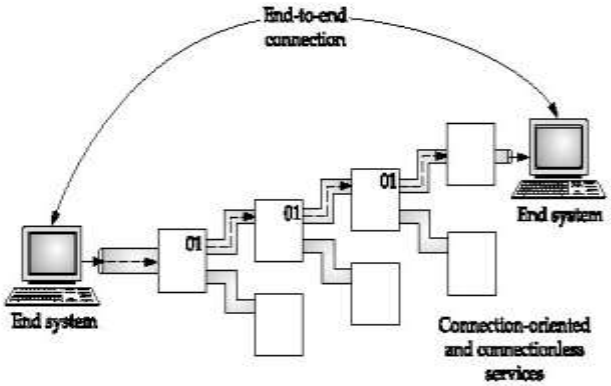


Figure 9.14: *Connection and Connectionless*

9.7.2 Connectionless:

Does not require a session connection between sender and receiver. The sender simply starts sending packets (called datagrams) to the destination. This service does not have the reliability of the connection-oriented method, but it is useful for periodic burst transfers. Neither system must maintain state information for the systems that they send transmission to or receive transmission from. A connectionless network provides minimal services.

9.8 Transmission Checks

Network data transmissions often produce errors, such as toggled, missing or duplicated bits. As a result, the data received might not be identical to the data transmitted, which is obviously a bad thing. Because of these transmission errors, network protocols very often use error-detection codes. Examples of error-detection codes include *parity checking*, *checksums* and *cyclic redundancy checks*.

9.8.1 Parity Checking

Parity checking refers to the process of using a parity bit to check that data has been transmitted accurately. A parity bit is an extra bit transmitted with a data unit that will be used to check its integrity. There are two types of parity: *odd parity* and *even parity*.

- ✓ In odd parity, the parity bit is added such that the total number of bits at 1, in the data unit, is an odd number.
- ✓ In even parity, the parity bit is added so that the total number of 1s is an even number.

7 bits of data	(count of 1 bits)	8 bits including parity	
		even	odd
0000000	0	0000000**0**	0000000**1**
1010001	3	1010001**1**	1010001**0**
1101001	4	1101001**0**	1101001**1**
1111111	7	1111111**1**	1111111**0**

Figure 9.15: *Parity Checking; Even and Odd Parity*

Example 1: What are the parity bits for the following data units in odd parity?

i) $\boxed{1}$11011011 ii) $\boxed{0}$01001010 iii) $\boxed{1}$10101010

Example 2: What are the parity bits for the following data units in even parity?

i) $\boxed{0}$11010100 ii) $\boxed{1}$11001011 iii) $\boxed{1}$00111000

9.8.2 Checksum

A **checksum** or **hash sum** is a small-size datum from a block of digital data for the purpose of detecting errors which may have been introduced during its transmission or storage. It is usually applied to an installation file after it is received from the download server. By themselves checksums are often used to verify data integrity, but should not be relied upon to also verify *data authenticity*.

The actual procedurewhich yields the checksum, given a data input is called a **checksum function** or **checksum algorithm**. Depending on its design goals, a good checksum algorithm will usually output a significantly different value, even for small changes made to the input. This is especially true of cryptographic hash functions, which may be used to detect many

data corruption errors and verify overall *data integrity*; if the computed checksum for the current data input matches the stored value of a previously computed checksum, there is a very high probability the data has not been accidentally altered or corrupted.

Checksum functions are related to *hash functions*, *fingerprints*, *randomization functions*, and *cryptographic hash functions*. However, each of those concepts has different applications and therefore different design goals. Checksums are used as cryptographic primitives in larger authentication algorithms.

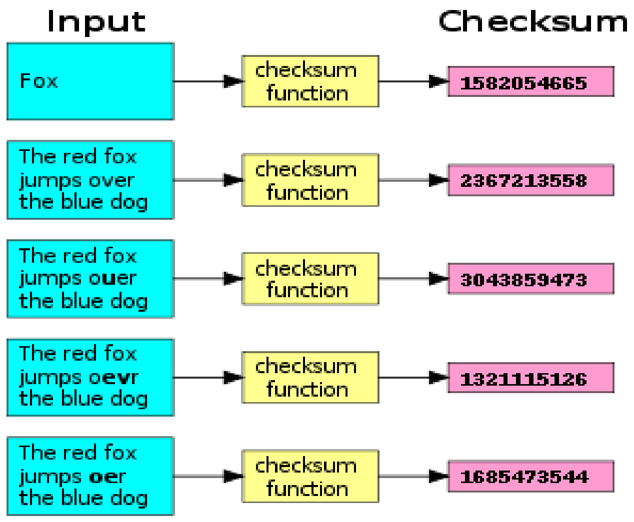


Figure 9.16: *Checksum function*

Check digits and parity bits are special cases of checksums, appropriate for small blocks of data (such as *Social Security numbers*, *bank account numbers*, *computer words*, *single bytes*, etc.). Some error-correcting codes are based on special checksums which not only detect common errors but also allow the original data to be recovered in certain cases.

9.8.3 Cyclic Redundancy Check

A CRC is an error-detection code in which each segment of the original message is combined with additional bits to make a binary number that is divisible by some previously chosen divisor.

i. ***K*** is the length of the message we want to send, i.e., the number of information bits.

ii. ***N*** is the total length of the message we will end up sending the information bits followed by the check bits. Peterson and Brown call this a code polynomial.

iii. ***n-k*** is the number of check bits. It is also the degree of the generating polynomial. The basic (mathematical) idea is that we're going to pick the n-k check digits in such a way that the code polynomial is divisible by the generating polynomial. Then we send the data, and at the other end we look to see whether it's still divisible by the generating polynomial; if it's not then we know we have an error, if it is, we hope there was no error.

9.8.4 Error Correcting Codes:

An error-correcting code is an algorithm for expressing a sequence of numbers such that any errors which are introduced can be detected and corrected (within certain limitations) based

on the remaining numbers. The study of error-correcting codes and the associated mathematics is known as *coding theory*.

Error detection is much simpler than error correction, and one or more "check" digits are commonly embedded in credit card numbers in order to detect mistakes. Early space probes like Mariner used a type of error-correcting code called a block code, and more recent space probes use convolution codes. Error-correcting codes are also used in CD players, high speed modems, and cellular phones. Modems use error detection when they compute checksums, which are sums of the digits in a given transmission modulo some number. The *ISBN* used to identify books also incorporates a check digit.

9.9 Peripheral Device Control

i) Buffers: A buffer is a data area shared by hardware devices or program processes that operate at different speeds or with different sets of priorities. The buffer allows each device or process to operate without being held up by the other. In order for a buffer to be effective, the size of the buffer and the algorithms for moving data into and out of the buffer need to be considered by the buffer designer. Like a cache, a buffer is a "midpoint holding place" but exists not so much to accelerate the speed of an activity as to support the coordination of separate activities.

This term is used both in programming and in hardware. In programming, buffering sometimes implies the need to screen data from its final intended place so that it can be edited or otherwise processed before being moved to a regular file or database.

ii) Interrupts: An interrupt is a signal from a device attached to a computer or from a program within the computer that causes the main program that operates the computer (the operating system) to stop and figure out what to do next. Almost all personal (or larger) computers today are ***interrupt-driven*** - that is, they start down the list of computer instruction s in one program (perhaps an application such as a word processor) and keep running the instructions until either (A) they can't go any further or (B) an interrupt signal is sensed. After the interrupt signal is sensed, the computer either resumes running the program it was running or begins running another program.

Basically, a single computer can perform only one computer instruction at a time. But, because it can be interrupted, it can take turns in which programs or sets of instructions that it performs. This is known as ***multitasking***. It allows the user to do a number of different things at the same time. The computer simply takes turns managing the programs that the user effectively starts. Of course, the computer operates at speeds that make it seem as though all of the user's tasks are being performed at the same time. (The computer's operating system is good at using little pauses in operations and user think time to work on other programs.)

An operating system usually has some code that is called an ***interrupt handler***. The interrupt handler prioritizes the interrupts and saves them in a queue if more than one is waiting to be handled. The operating system has another little program, sometimes called a *scheduler* that figures out which program to give control to next.

In general, there are ***hardware interrupts*** and ***software interrupts***. A **hardware interrupt** occurs, for example, when an I/O operation is completed such as reading some data into the computer from a tape drive. A **software interrupt** occurs when an application program terminates or requests certain services from the operating system. In a personal computer, a hardware *interrupt request (IRQ)* has a value associated with it that associates it with a particular device.

iii) Interrupt Latency (Priorities): Interrupt latency, also called interrupt response time, is the length of time that it takes for a computer interrupt to be acted on after it has been generated. In most computers, a trade-off exists among interrupt latency, throughput, and processor utilization.

Factors that affect interrupt latency include the microprocessor design (or architecture), the microprocessor clock speed, the particular OS employed, and the type of interrupt controller used. Minimum interrupt latency depends mainly on the configuration of the interrupt controller, which combines interrupts onto processor lines, and assigns priority levels to the interrupts. Maximum interrupt latency depends mainly on the OS.

iv) Polling : In electronic communication, 'polling' is the continuous checking of other programs or devices by one program or device to see what state they are in, usually to see whether they are still connected or want to communicate.

Specifically, in multipoint or multidrop communication (a controlling device with multiple devices attached that share the same line), the controlling device sends a message to each device, one at a time, asking each whether it has anything to communicate (in other words, whether it wants to use the line).

v) Handshaking: In telephone communication, handshaking is the exchange of information between two modems and the resulting agreement about which protocol to use that precedes each telephone connection. You can hear the handshaking in those crunching and other sounds when you make a dial-out call from your computer.

Since the modems at each end of the line may have different capabilities, they need to inform each other and settle on the highest transmission speed they can both use. At higher speeds, the modems have to determine the length of line delays so that echo cancellers can be used properly.

9.10 Communication Protocols

For proper communication in a network, different entities must speak the same language. There must be mutually acceptable conventions and rules about the content, timing and underlying mechanisms. These conventions and associated rules are referred as protocols.

Definition: *A protocol is a set of rules and conventions that govern how devices on a network communicate.*

The need for a protocol is obvious: it allows different computers from different vendors and with different operating characteristics to 'speak the same language'. The same protocols must be followed by each machine involved in the communication in order for the receiving host to be able to understand the message. A protocol may be physical or logical.

9.10.1 Physical Protocols

Physical protocols are concerned with how a device connects to a medium. They ensure that a device connected to a medium can transmit through the medium. They make sure that the layout of pins on the connectors is the same and that devices are correctly connected and configured. Few examples of physical protocols are 802.11 for Wi-Fi connections and DSL for broadband.

9.10.2 Logical Protocols

Logical protocols are concerned with data handling. They ensure that data are in the right format for the application, the bit rates match at both ends, and the same error correction is used. Examples of logical protocols are TCP/IP, HTTP, POP3, FTP, SMTP and WAP.

9.11 Backbone Networks

9.11.1 Definition:

Backbone part of the network is a core of the network with its high throughput capability and significant bandwidth. It is made for the ability of network to communicate with external networks (like Internet). It is a root of the network tree that has the rest of the network growing from it.

9.11.2 Types of Backbone Networks:

i) Serial Backbone Network:

Serial backbone is formed of two or more devices that are connected in a daisy chain (linked series). It is a simplest kind of backbone. As the one can see from *Figure 9.17*, serial backbone can be made not only from switches, but also from gateways and routers.

While designing the backbone, the one should consider the limit of the devices that can be connected to the backbone in the repeating fashion. Exceeding the limit would result in the unexpected errors and data loss in the network. Serial backbone networks are not very fault tolerant and not very scalable, that make them less commonly used that the distributed backbone.

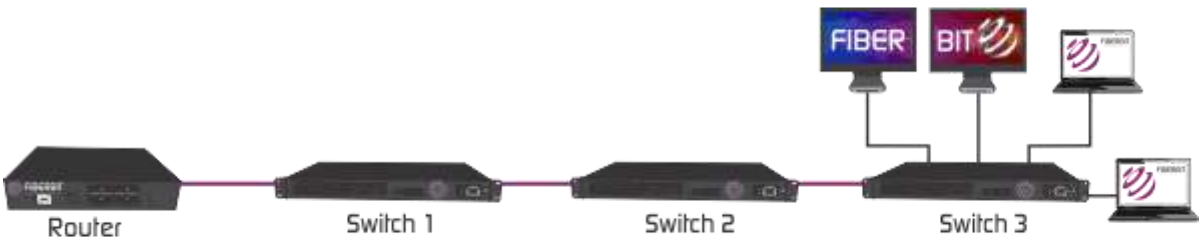


Figure 9.17: *Serial backbone*

ii) Distributed Backbone:

Distributed backbone uses hierarchical design of the network, where number of intermediate devices are connected to single or multiple connectivity devices. These central connectivity

devices could be switches or routers and shown with purple color in *Figure 18*. Overall, it is cheap, easy, and quick to implement the distributed backbone network.

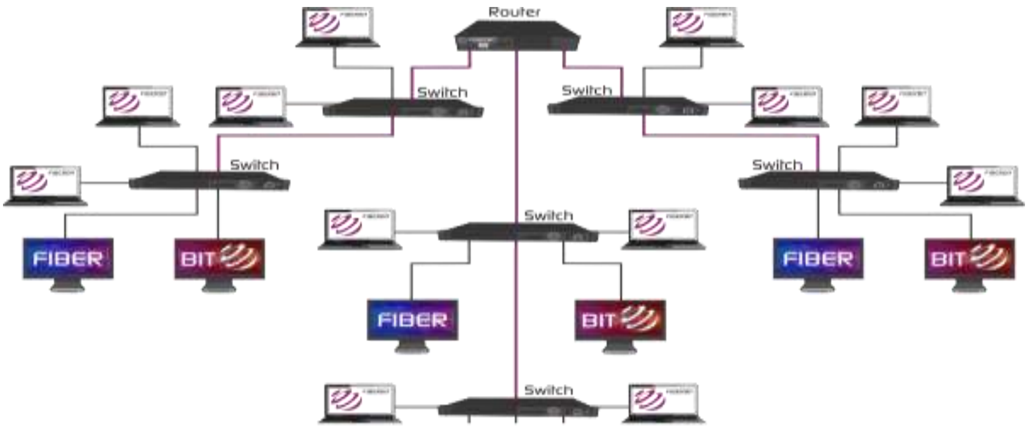


Figure 9.18: *Distributed Backbone*

iii) Collapsed Backbone:

This type of backbone uses single, powerful router as the central connection point for multiple sub networks. As *Figure 9.19* shows, the central device is the highest level of the backbone. It should have powerful computational power in order to manage big traffic coming in. This is highly risky, since if the central device fails, the whole network would be down. However, this type of backbone is useful for the one who wants to interconnect two types of sub networks, with ability to manage and troubleshoot them.

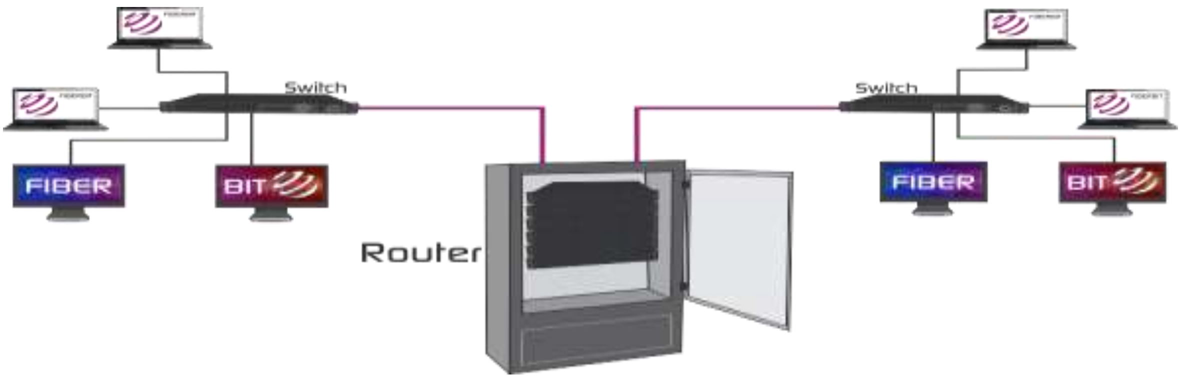


Figure 9.19: *Collapsed Backbone*

iv) Parallel Backbone:

Parallel backbone is a variation of the collapsed backbone, where devices are having more than one connection between them. As *Figure 9.20* shows, there are multiple connections between the high level routers and the network segments. Duplicate connections ensure networks availability at any time, higher speeds, and high fault tolerance. Logical drawback of this solutions is the increased price, since amount of required cabling is highly increased. It is not obligatory to have duplicate connections between all the devices, selective implementation of parallel structure would significantly lower the overall price and make additional ports of the devices available.

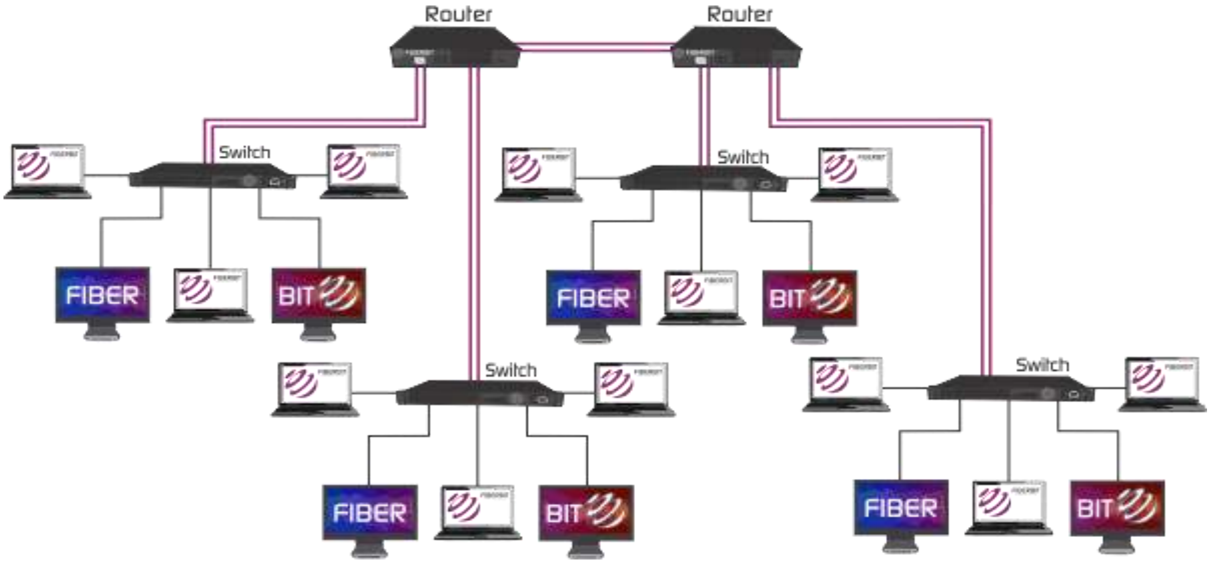


Figure 9.20: *Parallel Backbone*

9.12 COMPUTER NETWORKS

A computer network or data network is the interconnection of two or more computers to share resources (hardware, software and data) and communicate with one another. The connections (network links) between nodes are established using either cable media or wireless media. The best-known computer network is the Internet. Network computer devices that originate, route and terminate the data are called **network nodes**. Nodes can include hosts such as personal computers, phones, servers as well as networking hardware. Two devices are said to be networked together when one device is able to exchange information with the other device, whether or not they have a direct connection to each other. Computer networks support applications such as access to the *World Wide Web*, shared use of application and storage servers, printers, and fax machines, and use of email and instant messaging applications. Computer networks differ in the physical media used to transmit their signals, the communications protocols to organize network traffic, the network's size, topology and organizational intent.

9.12.1 Properties of Networks

A computer network has the following properties:

1. *Facilitates interpersonal communications*

People can communicate efficiently and easily via email, instant messaging, chat rooms, telephone, video telephone calls, and video conferencing.

2. *Allows sharing of files, data, and other types of information*

Authorized users may access information stored on other computers on the network. Providing access to information on shared storage devices is an important feature of many networks.

3. *Allows sharing of network and computing resources*

Users may access and use resources provided by devices on the network, such as printing a document on a shared network printer.

4. *May be insecure*

A computer network may be used by computer Crackers to deploy computer viruses or

computer worms on devices connected to the network, or to prevent these devices from accessing the network (denial of service).

5. *May be difficult to set up*

A complex computer network may be difficult to set up. It may be costly to set up an effective computer network in a large organization.

9.12.2 Types of Network

One way to categorize the different types of computer network designs is by their scope or scale. For historical reasons, the networking industry refers to nearly every type of design as some kind of *area network*. Common types of area networks are:

- LAN - Local Area Network
- WAN - Wide Area Network
- WLAN - Wireless Local Area Network
- MAN - Metropolitan Area Network
- SAN - Storage Area Network, System Area Network, Server Area Network, or sometimes Small Area Network.
- CAN - Campus Area Network, Controller Area Network, or sometimes Cluster Area Network
- PAN - Personal Area Network

LAN and WAN are the two primary and best-known categories of area networks, while the others have emerged with technology advances.

NB: *Network types differ from* network topologies *(such as bus, ring and star).*

i) LAN - Local Area Network: A LAN connects network devices over a relatively short distance. A networked office building, school, or home usually contains a single LAN, though sometimes one building will contain a few small LANs (perhaps one per room), and occasionally a LAN will span a group of nearby buildings. In TCP/IP networking, a LAN is often but not always implemented as a single IP subnet.

In addition to operating in a limited space, LANs are also typically owned, controlled, and managed by a single person or organization. They also tend to use certain connectivity technologies, primarily Ethernet and Token Ring.

ii) WAN - Wide Area Network: As the term implies, a WAN spans a large physical distance. The Internet is the largest WAN, spanning the Earth.A WAN is a geographically-dispersed collection of LANs. A network device called a router connects LANs to a WAN. In IP networking, the router maintains both a LAN address and a WAN address.

A WAN differs from a LAN in several important ways. Most WANs (like the Internet) are not owned by any one organization but rather exist under collective or distributed ownership and management. WANs tend to use technology like ATM, Frame Relay and X.25 for connectivity over the longer distances.

iii) LAN, WAN and Home Networking: Residences typically employ one LAN and connect to the Internet WAN via an Internet Service Provider (ISP) using a broadband modem. The ISP provides a WAN IP address to the modem, and all of the computers on the home network use LAN (so-called *private*) IP addresses. All computers on the home LAN can communicate directly with each other but must go through a central network gateway, typically a broadband router, to reach the ISP.

iv) Other Types of Area Networks

While LAN and WAN are by far the most popular network types mentioned, you may also commonly see references to these others:

- ***Metropolitan Area Network*** - a network spanning a physical area larger than a LAN but smaller than a WAN, such as a city. A MAN is typically owned and operated by a single entity such as a government body or large corporation.
- ***Campus Area Network*** - a network spanning multiple LANs but smaller than a MAN, such as on a university or local business campus.
- ***Storage Area Network*** - connects servers to data storage devices through a technology like Fibre Channel.
- ***System Area Network*** (*also known as Cluster Area Network*). Links high-performance computers with high-speed connections in a cluster configuration.

9.12.3 Wireless Networks

9.12.3.1 Introduction

A wireless network enables people to communicate and access applications and information without wires. This provides freedom of movement and the ability to extend applications to different parts of a building, city, or nearly anywhere in the world. Wireless networks allow people to interact with e-mail or browse the Internet from a location that they prefer.

Many types of wireless communication systems exist, but a distinguishing attribute of a wireless network is that communication takes place between computer devices. These devices include personal digital assistants (PDAs), laptops, personal computers (PCs), servers, and printers. Computer devices have processors, memory, and a means of interfacing with a particular type of network. Traditional cell phones don't fall within the definition of a computer device; however, newer phones and even audio headsets are beginning to incorporate computing power and network adapters. Eventually, most electronics will offer wireless network connections.

As with networks based on wire, or optical fiber, wireless networks convey information between computer devices. The information can take the form of e-mail messages, web pages, database records, streaming video or voice. In most cases, wireless networks transfer data, such as e-mail messages and files, but advancements in the performance of wireless networks is enabling support for video and voice communications as well.

9.12.3.2 Types of Wireless Networks

The types of wireless networks are defined on the bases of their size (that is the number of machines), their range and the speed of data transfer.

i) Wireless PAN – *Personal area network Wireless Personal Area Networks*

Such networks interconnect devices in small premises usually within the reach of a person for example invisible infra-red light and Bluetooth radio interconnects a headphone to a laptop by the virtue of WPAN. With the installation of Wi-Fi into customer electronic devices the Wi-Fi PANs are commonly encountered.

ii) Wireless LAN – *Local Area Network*

The simplest wireless distribution method that is used for interlinking two or more devices providing a connection to wider internet through an access point. OFDM or spread-spectrum technologies give clients freedom to move within a local coverage area while remaining connected to the LAN. LAN’s data transfer speed is typically 10 Mbps for Ethernet and 1 Gbps for Gigabit Ethernet. Such networks could accommodate as many as hundred or even one thousand users.

iii) Wireless MAN – *Metropolitan Area Networks*
Thewireless network that is used to connect at high speed multiple wireless LANs that are geographically close (situates anywhere in a few dozen kilometers). The network allows two or more nodes to communicate with each other as if they belong to the same LAN. The set up makes use of routers or switches for connecting with high-speed links such as fiber optic cables. WiMAX described as 802.16 standard by the IEEE is a type of WMAN.

iv) Wireless WAN

WAN is the wireless network that usually covers large outdoor areas. The speed on such network depends on the cost of connection that increases with increasing distance. The technology could be used for interconnecting the branch offices of a business or public internet access system. Developed on 2.4GHz band these systems usually contain access points, base station gateways and wireless bridging relays. Their connectivity with renewable source of energy makes them stand alone systems. The most commonly available WAN is internet.

v) Mobile devices networks

The advent of smart phones have added a new dimension in telecommunications, referred as “Mobile communication“; today’s telephones are not meant to converse only but to carry data.

a) GSM – *Global System for Mobile Communications* Global System for Mobile Communications is categorized as the base station system, the operation and support system and the switching system. The mobile phone is initially connected to the base system station

that establishes a connection with the operation and support station that later on connects to the switching station where the call is made to the specific user.

b) PCS – ***Personal Communications Service*** is a radio band that is employed in South Asia and North America; the first PCS service was triggered by Sprint.

c) D-AMPS - ***Digital Advanced Mobile Phone Service*** is the upgraded version of AMPS that is faded away due to technological advancements.

9.12.4 Network Topologies

Network topology is the layout or arrangement of the components of a network. It refers to the way in which computers and cables are connected together to build a network. Different types of topologies exist.

9.12.4.1 Bus Topology

In bus topology, all computers are connected to a single cable (trunk or backbone) known as bus, by a transceiver either directly or by using a short drop cable. Bus transmits in both directions such that any transmission can be received by all stations. All ends of the cable must be terminated, that is plugged into a device such as a computer or terminator, to avoid signals from bouncing back.

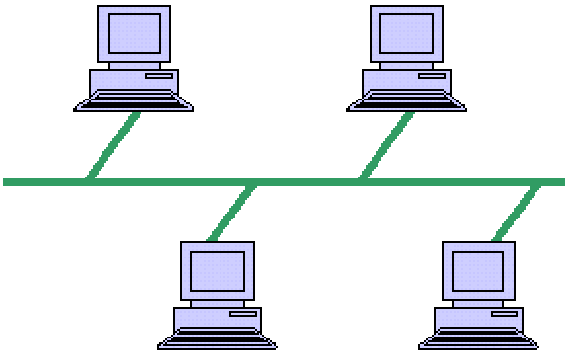


Figure 9.21: *Bus Topology*

a. Advantages

- ✓ Easy and inexpensive to set up as little cabling is required
- ✓ Easy to include additional stations without disrupting the network
- ✓ Failure of one node does not affect network

b. Disadvantages

- ✓ High rate of data collision
- ✓ Fails if there is any damage to the bus
- ✓ Any break in the bus is difficult to identify

9.12.4.2 Star Topology

In a star topology, all the computers are connected to a central device which could be a computer, a hub or a switch. Any communications between computers in this topology must pass through the central node. As such, the central node controls all the activities of the network.

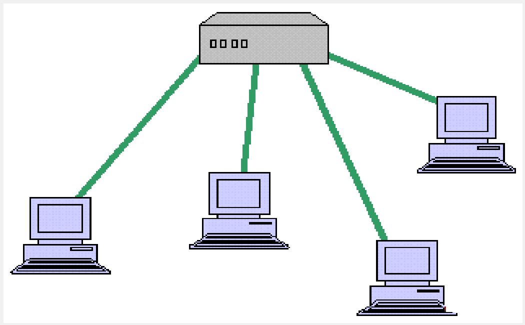


Figure 9.22: *Star Topology*

a. **Advantages**
- ✓ Breakdown of a node does not affect the network
- ✓ No disruption of the network when connecting or removing devices
- ✓ It is easy to detect faults

b. **Disadvantage**
- ✓ Failure of the central node affects the entire network
- ✓ It is costly due to the amount of cables required to connect the devices

9.12.4.3 Ring Topology

In ring topology, all the nodes are connected in the form of a closed loop such that each node is connected to two others. It uses an empty data packet called a token and a special protocol called token ring. Packets travel around the ring in a clockwise direction. To transmit, a node requires an empty token.

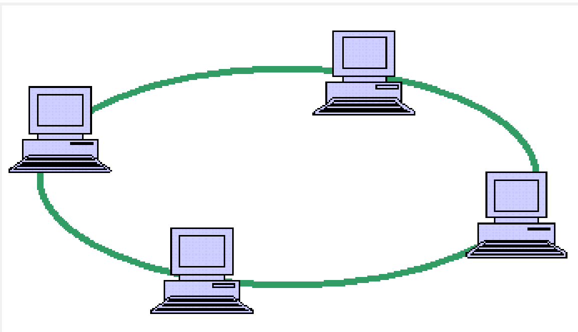


Figure 9.23: *Ring Topology*

a. **Advantage**
- ✓ No collision as a station needs the token to transmit
- ✓ Each computer acts like a repeater so signals are not attenuated

b. Disadvantage

- ✓ If a node in the network fails, the entire network fails
- ✓ Network is disrupted when additional stations are added

9.12.4.4 Tree Topology

A tree topology joins multiple star topologies together onto a bus. In its simplest form, only hub devices connect directly to the tree bus, and each hub functions as the root of a tree of devices. This bus/star hybrid approach supports future expansion of the network much better than a bus (limited in the number of devices due to the broadcast traffic it generates) or a star (limited by the number of hub connection points) alone.

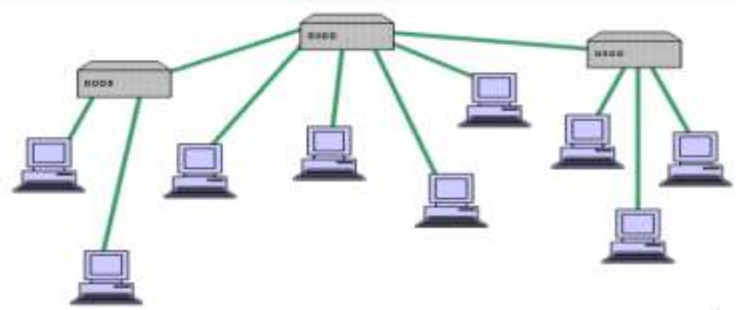


Figure 9.24: *Tree Topology*

9.12.4.5 Mesh Topology

Mesh topology introduces the concept of routes. Unlike each of the previous topologies, messages sent on a mesh network can take any of several possible paths from source to destination. (Recall that even in a ring, although two cable paths exist, messages can only travel in one direction.) Some WANs, most notably the Internet, employ mesh routing.

A mesh network in which every device connects to every other is called a full mesh. As shown in the illustration besides, partial mesh networks also exist in which some devices connect only indirectly to others.

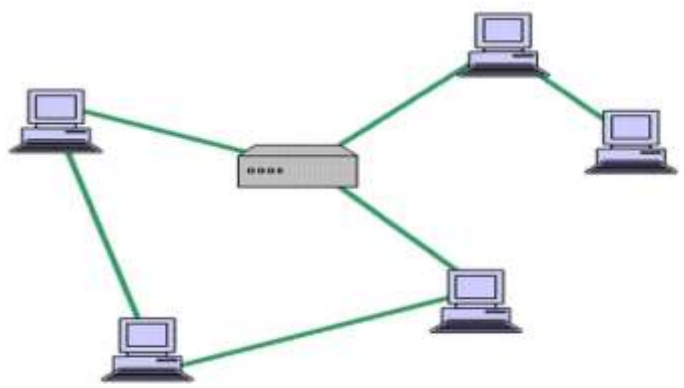


Figure 9.25: *Mesh Topology*

9.12.5 Network Standards

9.12.5.1 Ethernet Network

Ethernet (IEEE 802.3 standard) is the most common and widely used technology to establish a local area network. An Ethernet network is formed by physically connecting the individual computer units to each other in a bus topology or a star topology. Ethernet's media access policy is CSMA/CD (Carrier Sense Multiple Access with Collision Detection).

- ✓ **CS**: means that a station listens to (senses) the medium and transmits only if medium is idle
- ✓ **MA**: means that any station can use (access) the medium
- ✓ **CD**: means that each station stops transmitting immediately it senses a collision

When a collision is detected, the two stations involved will retransmit after a random time wait created by a back off algorithm.

9.12.5.2 Token Ring Network

Token ring (IEEE 802.5 standard) is a network technology developed by IBM in which computers are connected together in a ring. Token ring's media-access method is called token passing. A special message, called token, circulates along the ring from one computer to another and each computer can transmit only while it is holding the token. Information flows in one direction along the ring from source to destination and back to source. When a station wishes to transmit, it waits for the empty token to pass by. It seizes it and inserts data into it and then releases it to the medium. The token circulates until it gets to the destination computer that picks it and retrieves the data. After retrieving the data, it regenerates the token and sends it back to the medium.

9.12.5.3 Fiber Distributed Data Interface

FDDI is a network technology that uses fiber-optic cables in a ring topology with dual rings on which information can travel in opposite directions. The media access method for FDDI is token passing. The primary ring is used for data transmission, and the secondary ring remains idle. Because of this double ring topology, if a station fails or a cable becomes damaged, the dual ring is automatically wrapped around itself, forming a single ring. This prevents downtime as a result of a failed machine or faulty wiring.

9.12.5.4 Wireless Network Standards

Wireless networks are established without physical wiring techniques involved. They use radio and infrared signals and are based around one of these technologies: *Bluetooth*, *Wi-Fi*, *WiMax*, *terrestrial microwaves* and *satellite*.

a. Bluetooth

Bluetooth is a low power, short-range wireless technology largely used to interconnect computing devices into a personal area network. It is based on IEEE standard 802.15 which gives specifications for Wireless Personal Area Network (WPAN).

b. Wi-Fi

Wi-Fi stands for Wireless Fidelity. It is based on a set of wireless networking technologies known as 802.11. These include 802.11b, 802.11a, 802.11g and 802.11n. The range of Wi-Fi network transmission is about 30-40m indoors and up to about 100m outdoors.
The table below shows the different 802.11 standards for wireless networking.

Specification	Popular name	Frequency	Speed	Compatible with
802.11a	Wireless-A	5 GHz	54 Mbps	
802.11b	Wireless-B	2.4 GHz	11 Mbps	Wireless-B
802.11g	Wireless-G	2.4 GHz	54 Mbps	Wireless-B, -G
802.11n	Wireless-N	2.4 GHz	100 Mbps	Wireless-B, -G, -N

c. WiMax

WiMax stands for Worldwide Interoperability for Microwave Access. It is based on IEEE standard 802.16 and facilitates high speed wireless network links to both fixed and mobile devices. The range of a WiMax wireless connection is around 3-10km.WiMax service providers are now just entering the market, offering customers an alternative to a DSL Internet connection.

9.12.6 Network Architectures

9.12.6.1 Client/Server Architecture

Client/server is a network architecture in which a more powerful computer called server is dedicated to serving less powerful computers called clients. Servers hold shared resources like files, programs and the network operating system. They provide access to network resources to all the users of the network. There are many different kinds of servers, and one server can provide several functions. For example, there are file servers, print servers, mail servers, database servers and Web servers. Users run applications on client workstations which rely on servers for resources such as files, devices and even processing power.
Internet services are organized according to a client/server architecture. Client programs, such as Web browsers and file transfer programs create connections to servers, such as Web and FTP servers. The clients make requests and the server responds to the requests by providing the services requested by the client.

9.12.6.2 Peer-to-Peer Architecture

Peer-to-peer (P2P) is a network configuration in which all the workstations (computers) have equal capabilities and responsibilities. Each workstation acts both as a server and a client. This means that any computer on the network can provide services to any other computer. Peer-2-peer is usually implemented where strict security is not necessary. P2Pnetworks are generally simpler and less expensive, but they usually do not offer the same performance under heavy loads.

Remark *A hybrid network combines client/server and peer-to-peer architectures. It is the most commonly used network architecture.*

9.12.7 Computer Network Components

All types of computer networks require special networking software and hardware to allow different computers to communicate with each other. The most important software component required for a network is the network operating system (NOS) while there are many types of hardware devices which are either installed or connected to the computer terminals in order to construct a network.

9.12.7.1 Network Operating System

A network operating system is an operating system which includes networking features. It contains special functions, protocols and device drivers that enable the computer to be connected to a network. NOS provide the ability to share resources and the ability to manage a network name directory, security, and other housekeeping aspects of a network.
Examples of network operating systems are *Windows-NT*, *Windows-2000 server*, *Windows server 3000, Novell Netware* and *Artisoft LANstatic*.
Some multi-purpose operating systems like *Windows XP*, *Windows 7* and *Mac OS 10*, come with capabilities that enable them to be described as network operating systems.

9.12.7.2 Network Interface/Adapter Card

A network interface card (NIC) provides the physical interface (link) between the computer and the communication medium. A NIC manages the communication and network protocol for the PC.It prepares data, sends data and controls the flow of data. It plugs into the system board and provides ports for connection to the network. A NIC is also called a LAN card or network adapter card. There are two kinds of NIC: wired NIC for wired networks and wireless NIC (WNIC) for wireless networks. A NIC may be designed as an Ethernet card, a Token Ring card, or an FDDI card (but not all three).

9.12.7.3 Hub

A hub is a device that works as central connecting point for multiple computers in a network. It has ports to which the computers in the network are connected. Data sent to the hub is broadcasted to all the ports but, only the destination computer receives it. There are three kinds of hubs:

i. **Passive hubs** which only split the transmission signal so it can be sent to all the ports
ii. **Active hubs** (also called Multiport Repeaters) which regenerate data bits to maintain a strong signal over extended cable lengths
iii. **Intelligent hubs** (also called Concentrators) usually have their own microprocessor chips and network operating systems. They can be managed remotely on the network.

9.12.7.4 Switch

A switch is used at the same place as a hub but the difference between the two is that a switch has a switching table within it. A switching table stores the Media Access Control (MAC) address of every computer connected to the switch and sends the data only to the requested address, unlike the hub which broadcasts the data to all the ports. Switches can therefore be considered as an advanced form of hubs.
A MAC address is a built-in number (i.e. set by the manufacturer) consisting of 12 hexadecimal digits that uniquely and permanently identifies the network adapter of a computer. Examples of

a MAC addresses are 00-14-22-DA-67-15 and 00-13-02-31-E8-BA.MAC address is also called the *physical address*. Under Windows, the MAC address of a computer can be displayed by typing `ipconfig /all` at a Command prompt.

9.12.7.5 Repeater

A repeater is a device used to expand the boundaries of a wired or wireless network. With physical media, data transmissions can only span a limited distance before the quality of the signal degrades. Repeaters are used to preserve signal integrity and extend the distance over which data can safely travel by regenerating the signals they receive. Active hubs are considered as repeaters (multiport repeaters).

9.12.7.6 Bridge

A bridge, also called a layer 2 switch, is a device used to create a connection between two separate computer networks or to divide one network into segments. Creating multiple segments in a local network reduces the network traffic making the network to be faster and more efficient. A bridge performs its segmenting function by examining the data packet and forwarding it to other physical segments only if necessary.

9.12.7.7 Router

A router is a device that joins several networks together and is responsible for routing data from one network to another. It keeps track of the IP addresses of the computers on the networks connected to its network interface cards and directs data packets appropriately. It is more powerful than a bridge because instead of just choosing network segments based on previous traffic, a router can look up the best route for a packet to take. Routers can be computers with operating systems and special network software, or they can be other dedicated devices built by network manufacturers. The Internet relies heavily on routers.

9.12.7.8 Modem

A modem (modulator/demodulator) is a device that encodes data for transmission over a particular medium, such as telephone line, coaxial cable, fiber optics, or microwaves. It converts digital signals from a computer to analog signals or waveform for transmission over a medium (*modulation*) and converts analog signals from the medium to digital signals understandable by the computer (*demodulation*).
Common types of modems are:

- ✓ Dial-up Modem
- ✓ Cable Modem
- ✓ DSL Modem
- ✓ Sat modem

9.12.7.9 Multiplexer

A multiplexer abbreviated MUX, is a device that takes input signals from different sources and transmits them over a single transmission line. This process is known as multiplexing. There are different types of multiplexing:

- ✓ *Frequency-division multiplexing (FDM),* in which the carrier bandwidth is divided into sub channels of different frequency widths, each carrying a signal at the same time in parallel.

- ✓ *Time-division multiplexing (TDM*), in which the multiple signals are carried over the same channel in alternating time slots.
- ✓ *Code-division multiplexing (CDM)*, in which the multiple signals are carried over the same channel but every signal is coded differently.

In Cameroon, telecommunication companies like ORANGE, NEXTEL and MTN use a combination of FDM and TDM called **GSM** while CAMTEL uses CDMA.

9.12.7.10 Cables

Cables are used to link computers in a LAN. There are three types of cables commonly used:

- ✓ Coaxial cable
- ✓ Twisted pair cable
- ✓ Fiber optic cable

9.12.7.11 Gateway

A gateway is a device that connects two dissimilar computer networks using direct and systematic translation between protocols. A gateway translates outgoing network traffic to the protocol needed by the destination network. The term gateway is also sometimes loosely used to describe any device that acts as the entry or exit point for a network.

9.13 Network Security

Network security consists of provisions and policies adopted by the network administrator to prevent and monitor unauthorized access, misuse, modification, or denial of the computer network and its network-accessible resources. Network security is the authorization of access to data in a network, which is controlled by the network administrator. Some methods used to keep unauthorized persons out of a network include:

i) User ID*:* A user ID is a number or name that is unique to a person using the network. The network administrator uses this to allocate file space for each user’s work. It is also used to permit a user to access certain files.

ii) Password*:* A password is a string of characters (letters and/or numbers) that the user or network administrator uses to authenticate the user to the system. The user ID is shown on the screen while a password is hidden. You are only allowed access to the network if you type in the right password.

iii) Firewall*:* A firewall is a network device and/or software for controlling network security and access rules. Firewalls are typically configured to reject access requests from unrecognized sources while allowing actions from recognized ones. The vital role firewalls play in network security grows in parallel with the constant increase in cyber-attacks.

9.14 Benefits and Limitations of Data Communication

9.14.1 Benefits

- ✓ Sharing devices such as printers saves money.
- ✓ Site (software) licenses are likely to be cheaper than buying several standalone licenses.
- ✓ Files can easily be shared between users.
- ✓ Network users can communicate by email and instant messaging.
- ✓ Data is easy to backup as all the data is stored on the file server.

- ✓ Organizations can organize videoconferences (videoconferencing)
- ✓ Employees can work from home (telecommuting)

9.14.2 Limitations

- ✓ Purchasing the network cabling and file servers can be expensive.
- ✓ Managing a large network is complicated. It requires training and a network manager usually needs to be employed.
- ✓ If the file server breaks down the files on the server become inaccessible. Email might still work if it is on a separate server. The computers can still be used but are isolated.
- ✓ Viruses can easily spread to other computers throughout the network, if one computer is infected..
- ✓ There is a danger of hacking, particularly with wide area networks. Security procedures are needed to prevent such abuse, eg a firewall.

Drill Questions:

1. Explain the following modes of data flow in data communication systems
a) Simplex channel b) Half Duplex c) Full Duplex

2.(i)(a) What is internetworking? Define the term *The Internet,* (**Q9, CGCE 2014**)
(b) Describe the terms; Intranet and Extranet.
(ii) Explain what each of the following hardware is used for in a computer network.
a) Switch b) Repeater c) Router d) Multiplexer and demultiplexer

3.(i)a) Differentiate between Physical network topology and logical network topology
b) Name and illustrate ANY TWO of the three main types of network topologies. For each topology, give one protocol used, one advantage and one disadvantage.
c) Differentiate between routers and gateways (**CSC Q2, CGCE 2013**)

5.a) What is a network topology?
b) Explain two considerations in selecting a topology for a school network.
(c) With the help of well labelled diagrams, describe three commonly used standard network topologies, giving one advantage and one disadvantage for each use. (**CGCE 2015**)

6(a) Explain the main difference between peer to peer and client/server networks.
(b) Give two disadvantages of the client/server network. (**Q2i, CGCE 2016**)

CHAPTER 10: INTERNET

10.1 Introduction

Internet is defined as an Information super Highway, to access information over the web. However, Internet is a world-wide global system of interconnected computer networks.Internet uses the standard Internet Protocol (TCP/IP).Every computer in internet is identified by a unique IP address.IP Address is a unique set of numbers (such as 110.22.33.114) which identifies a computer location.

A special computer DNS (Domain Name Server) is used to give name to the IP Address so that user can locate a computer by a name.For example, a DNS server will resolve a name **http://www.ubuea.cm** to a particular IP address to uniquely identify the computer on which this website is hosted.Internet is accessible to every user all over the world.

10.2 Brief History of Internet

Many years ago, the military of the United States of America desired to interconnect or link their computers in order to better understand and manage information and communication with respect to enemy attacks in times of crisis. In the year 1969 the Department of Defense (DoD) then developed an experimental network called the Advanced Research Project Agency Network (ARPANet)

In the year 1980, the National Science Foundation of the United States of America then developed the technology of ARPANet to produce the National Science Foundation Network (NSFNet) which now enabled universities and other school establishments in the USA to be interconnected. After a great deal of work, a network which enabled the transfer of large amounts of information at very high speed which is today called the *Internet* was developed.

The Internet can be defined as a worldwide/global system of interconnected computer networks. It is the network of networks in which users can view information on the World Wide Web, exchange electronic mail, participate in electronic discussion forums (newsgroups), send files from any computer to any other and even use each other’s computers directly if they have appropriate passwords. Another name for the Internet is information superhighway.

10.3 Advantages and Disadvantages of Internet

10.3.1 Advantages of Internet

a) Internet allows us to communicate with the people sitting at remote locations. There are various apps available on the web that uses Internet as a medium for communication. One can find various social networking sites such as: Facebook, Twitter, Yahoo, Google+, Flickr, Orkut

b) One can surf for any kind of information over the internet. Information regarding various topics such as Technology, Health & Science, Social Studies, Geographical Information, Information Technology, Products etc. can be surfed with help of a search engine.

c) Apart from communication and source of information, internet also serves a medium for entertainment. Following are the various modes for entertainment over internet; Online Television, Online Games, Songs, Videos, Social Networking Apps.

d) Internet allows us to use many services like: Internet Banking, Matrimonial Services, Online Shopping, Online Ticket Booking, Online Bill Payment, Data Sharing, E-mail

e) Internet provides concept of *electronic commerce* that allows the business deals to be conducted on electronic systems

10.3.2 Disadvantages of Internet

a) There are always chances to loose personal information such as name, address, credit card number. Therefore, one should be very careful while sharing such information. One should use credit cards only through authenticated sites.

b) Another disadvantage is the ***Spamming***. Spamming corresponds to the unwanted e-mails in bulk. These e-mails serve no purpose and lead to obstruction of entire system.

c) Virus can easily be spread to the computers connected to internet. Such virus attacks may cause your system to crash or your important data may get deleted.

d) Also a biggest threat on internet is pornography. There are many pornographic sites that can be found, letting your children to use internet which indirectly affects the children healthy mental life.

e) There are various websites that do not provide the authenticated information. This leads to misconception among many people.

10.4 Intranet, Extranet their Advantages and Disadvantages

10.4.1 Intranet

Intranet is defined as private network of computers within an organization with its own server and firewall. Moreover we can define Intranet as a system in which multiple PCs are networked to be connected to each other. PCs in intranet are not available to the world outside of the intranet.Usually each company or organization has their own Intranet network and members/employees of that company can access the computers in their intranet.Every computer in internet is identified by a unique IP address.Each computer in Intranet is also identified by an IP Address, which is unique among the computers in that Intranet.

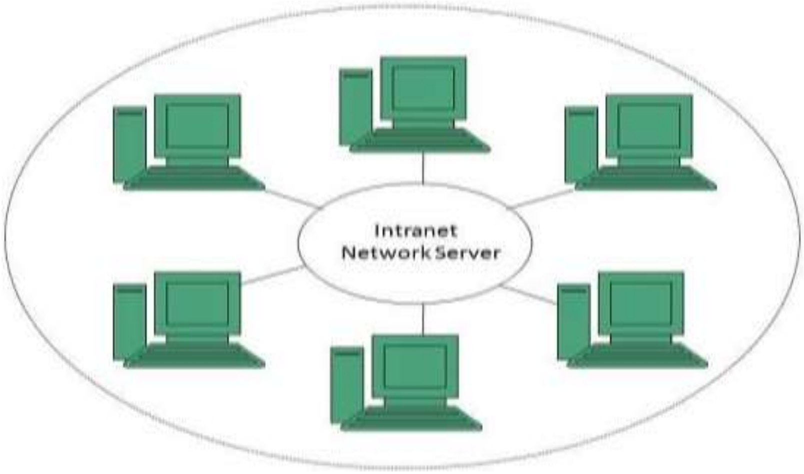


Figure 10.1: *Architecture of an Intranet*

a) Advantages of Intranets

Intranet is very efficient and reliable network system for any organization. It is beneficial in every aspect such as collaboration, cost-effectiveness, security, productivity and much more.

i) **Communication:** Intranet offers easy and cheap communication within an organization. Employees can communicate using chat, e-mail or blogs.

ii) **Time Saving:** Information on Intranet is shared in real time.

iii) **Collaboration:** Information is distributed among the employees as according to requirement and it can be accessed by the authorized users, resulting in enhanced teamwork.

iv) **Platform Independency:** Intranet can connect computers and other devices with different architecture.

v) **Cost Effective:** Employees can see the data and other documents using browser rather than printing them and distributing duplicate copies among the employees, which certainly decreases the cost.

vi) **Workforce Productivity:** Data is available at every time and can be accessed using company workstation. This helps the employees work faster.

vii) **Business Management:** It is also possible to deploy applications that support business operations.

viii) **Security:** Since information shared on intranet can only be accessed within an organization, therefore there is almost no chance of being theft.

ix) **Specific Users:** Intranet targets only specific users within an organization therefore, once can exactly know whom he is interacting.

x) **Immediate Updates:** Any changes made to information are reflected immediately to all the users.

b) Limitations of Intranets

i) Management does need to stop control of specific information, this problem can be minimized but with appropriate prudence.
ii) The other disadvantage of Intranet is security issue such as unauthorized access, Denial of service attack, Packet sniffing.
iii) Intranet gathered everything in one location which is really good but if it is not prearranged then you will spoil everything.
iv) The cost of intranet is very high but has lots of advantages after implementing.

c) Applications of intranet

i) **Document publication applications:** Document publication applications allow publishing documents such as manuals, software guide, employee profits etc without use of paper.

ii) **Electronic resources applications:** It offers electronic resources such as software applications, templates and tools, to be shared across the network.

iii) **Interactive Communication applications:** Like on internet, we have e-mail and chat like applications for Intranet, hence offering an interactive communication among employees.

iv) **Support for Internet Applications:** Intranet offers an environment to deploy and test applications before placing them on Internet.

e) Intranet Vs Internet

Apart from similarities there are some differences between the two. The following are the differences between Internet and Intranet:

Intranet	Internet
Localized network	Worldwide Network
Doesn't have access to internet	Have access to internet
More expensive	Less Expensive
More Safe	Less Safe
More reliability	Less Reliability

Table 10.1: *Difference between Intranet and Internet*

10.4.2 Extranet

Extranet refers to network within an organization, using internet to connect to the outsiders in controlled manner. It helps to connect businesses with their customers and suppliers and therefore allows working in a collaborative manner.

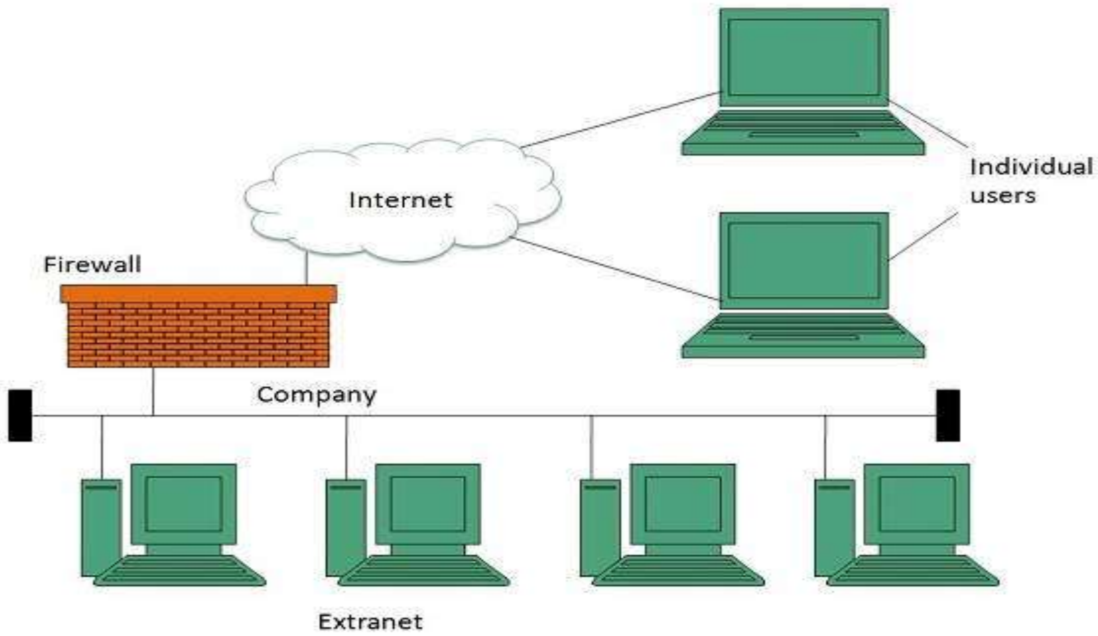


Figure 10.2: *Architecture of an Extranet*

a) Benefits of Extranets

Benefits can depend to a large degree on your reasons for introducing the extranet in the first place. However, the types of benefits that organizations using extranets typically experience include:

i. More ***Integrated Supply Chains*** Through The Use Of Online Ordering, Order Tracking And Inventory Management
ii. ***Reduced Costs*** By Making Manuals And Technical Documentation Available Online To Trading Partners And Customers
iii. ***More Effective Collaboration*** Between Business Partners - Perhaps Members Of A Project Team - By Enabling Them To Work Online On Common Documentation
iv. ***Improved Business Relationships*** With Key Trading Partners Because Of The Close Collaborative Working That Extranets Support
v. ***Improved Customer Service*** By Giving Customers Direct Access To Information And Enabling Them To Resolve Their Own Queries
vi. ***A Single User Interface*** Between You And Your Business Partners
vii. Improving The **Security** Of Communications Between You And Your Business Partners, Since Exchanges Can Take Place Under A Controlled And Secure Environment
viii. ***Shared News*** Of Product Development Exclusively With Partner Companies
ix. ***Flexible Working*** For Your Own Staff, As An Extranet Allows Remote And Mobile Staff To Access Core Business Information 24 Hours A Day, Irrespective Of Location

b) Limitations of Extranets

i) **Hosting:** Where the extranet pages will be held i.e. who will host the extranet pages. In this context there are two choices:

- Host it on your own server.
- Host it with an Internet Service Provider (ISP) in the same way as web pages.

But hosting extranet pages on your own server requires high bandwidth internet connection which is very costly.

ii) **Security**: Additional firewall security is required if you host extranet pages on your own server which result in a complex security mechanism and increase work load.

iii) **Accessing Issues**: Information cannot be accessed without internet connection. However, information can be accessed in Intranet without internet connection.

iv) **Decreased Interaction**: It decreases the face to face interaction in the business which results in lack of communication among customers, business partners and suppliers.

d) Extranet Vs Intranet

Extranet	Intranet
Internal network can be accessed externally	Internal network cannot be accessed externally
Extranet is extension of company's intranet	Only limits users of a company
For limited external communication between customers, suppliers and business partners	Only for communication within a company

Table 10.2: *Difference between Intranet and Extranet*

10.5 Reference Model

Reference Model offers a means of standardization which is acceptable worldwide. Since people using the computer network are located over a wide physical range and their network devices might have heterogeneous architecture. In order to provide communication among heterogeneous devices, we need a standardized model i.e. a reference model, which would provide us way how these devices can communicate regardless their architecture.

We have two reference models such as **OSI** model and **TCP/IP** reference model, however, the OSI model is a hypothetical one but the TCP/IP is absolutely practical model.

10.5.1 OSI Model

OSI is acronym of **Open System Interface**. This model is developed by the **International organization of Standardization (ISO)** and therefore also referred as **ISO-OSI** Model.

The OSI model consists of seven layers as shown in the following diagram. Each layer has a specific function, however each layer provide services to the layer above.

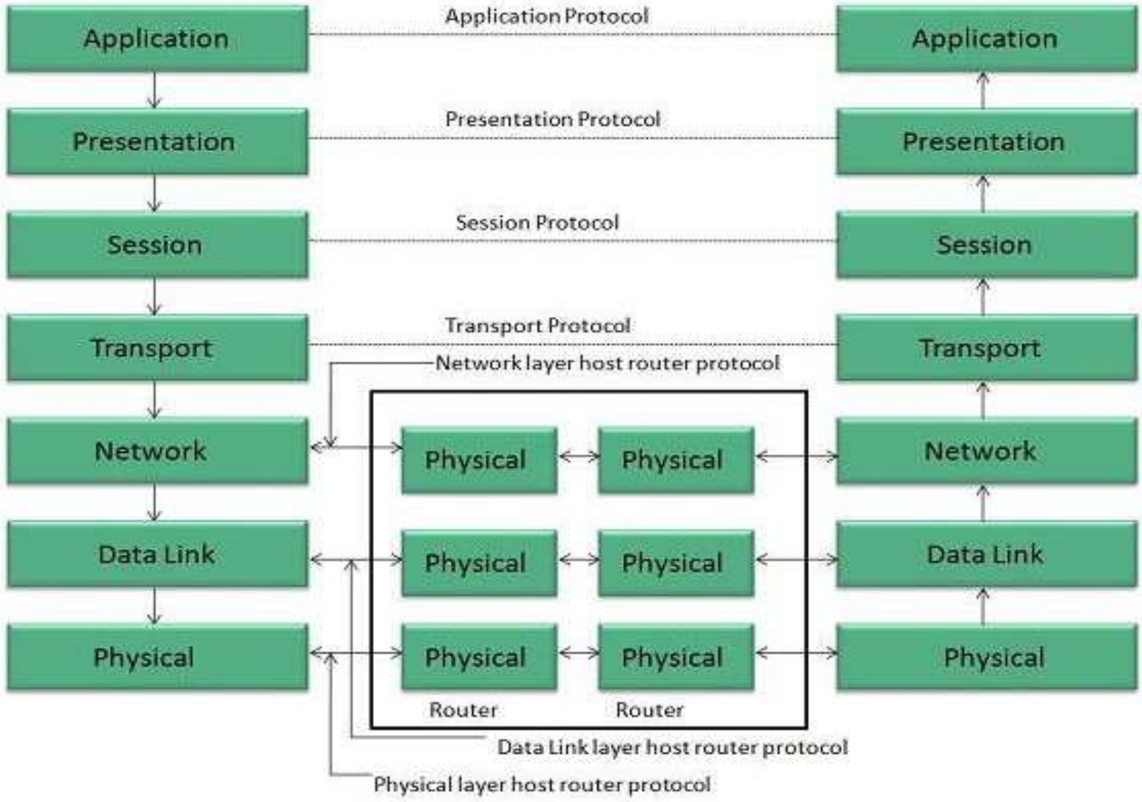


Figure 10.3: *OSI Model*

i) Physical Layer: The Physical layer is responsible for the following activities:

- Activating, maintaining and deactivating the physical connection.
- Defining voltages and data rates needed for transmission.
- Converting digital bits into electrical signal.
- Deciding whether the connection is simplex, half duplex or full duplex.

ii) Data Link Layer: The data link layer performs the following functions:

- Performs synchronization and error control for the information which is to be transmitted over the physical link.
- Enables error detection, and adds error detection bits to the data which are to be transmitted.

iii) Network Layer: Following are the functions of Network Layer:

- To route the signals through various channels to the other end.
- To act as the network controller by deciding which route data should take.
- To divide the outgoing messages into packets and to assemble incoming packets into messages for higher levels.

iv) Transport Layer: The Transport layer performs the following functions:

- It decides if the data transmission should take place on parallel paths or single path.
- It performs multiplexing, splitting on the data.
- It breaks the data groups into smaller units so that they are handled more efficiently by the network layer.

The Transport Layer guarantees transmission of data from one end to other end.

v) Session Layer: The Session layer performs the following functions:

- Manages the messages and synchronizes conversations between two different applications.
- It controls logging on and off, user identification, billing and session management.

vi) Presentation Layer: The Presentation layer performs the following functions:

- This layer makes it sure that the information is delivered in such a form that the receiving system will understand and use it.

vii) Application Layer: The Application layer performs the following functions:

- It provides different services such as manipulation of information in several ways, retransferring the files of information, distributing the results etc.
- The functions such as LOGIN or password checking are also performed by the application layer.

10.5.2 TCP/IP MODEL

TCP/IP model is practical model and is used in the Internet. TCP/IP is acronym of Transmission Control Protocol and Internet Protocol.

The **TCP/IP** model combines the two layers (Physical and Data link layer) into one layer i.e. **Host-to-Network** layer. The following diagram shows the various layers of TCP/IP model:

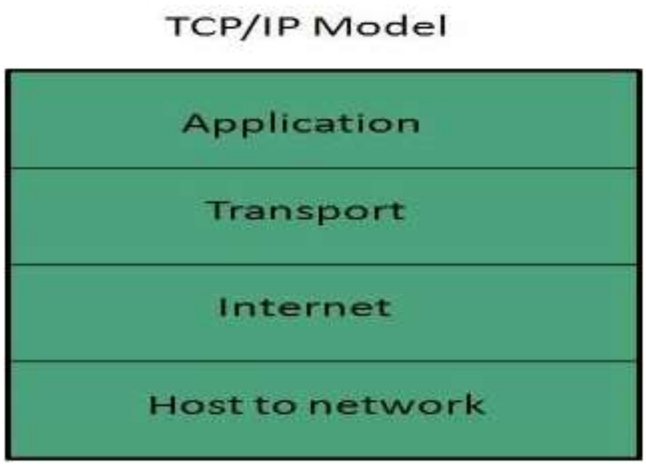


Figure 10.4: TCP/IP Model

i) Application Layer: This layer is same as that of the OSI model and performs the following functions:

- It provides different services such as manipulation of information in several ways, retransferring the files of information, distributing the results etc.
- The functions such as LOGIN or password checking are also performed by the application layer.

Protocols used: *TELNET, FTP, SMTP, DN, HTTP, NNTP (Network News Transfer Protocol)* are the protocols employed in this layer.

ii) Transport Layer: It does the same functions as that of transport layer in OSI model. Here are the key points regarding transport layer:

- It uses *TCP* and *UDP (User Datagram Protocol***)** protocol for end to end transmission.
- TCP is reliable and *connection oriented protocol***.**
- TCP also handles flow control.
- The UDP is not reliable and a connection less protocol also does not perform flow control.

Protocols used: *TCP/IP and UDP* protocols are employed in this layer.

iii) Internet Layer: The function of this layer is to allow the host to insert packets into network and then make them travel independently to the destination. However, the order of receiving the packet can be different from the sequence they were sent.

Protocols used: *Internet Protocol (IP)* is employed in Internet layer.

iv) Host-to-Network Layer: This is the lowest layer in TCP/IP model. The host has to connect to network using some protocol, so that it can send IP packets over it. This protocol varies from host to host and network to network.

Protocols used: *ARPANET, SATNET, LAN, packet radio* are the protocols which are used in this layer.

10.6 Internet Connectivity

10.6.1 Internet Service Provider

Internet Service Provider (ISP) is a company offering access to internet. They offer various services: Internet Access, Domain name registration, Dial-up access, Leased line access

10.6.2 ISP Types

ISPs can broadly be classified into six categories as shown in the following diagram:

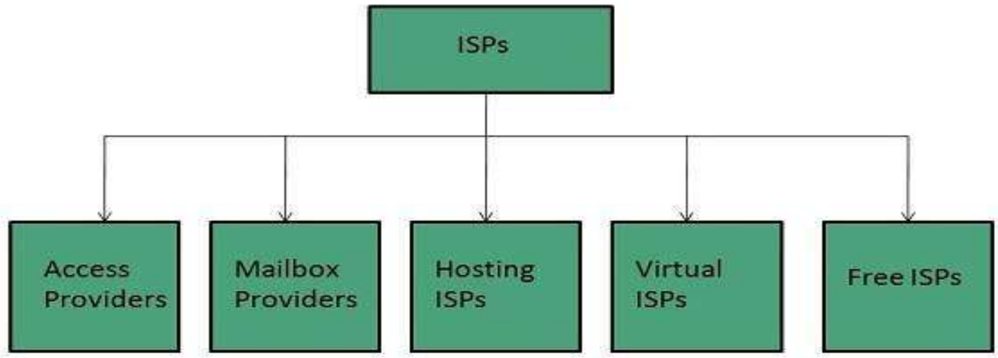


Figure 10.5: *ISP types*

a) ***Access Providers*****:** They provide access to internet through telephone lines, cable wi-fi or fiber optics.

b) ***Mail Box Providers*****:** Such providers offer mailbox hosting services.

c) ***Hosting ISPs*****:** Hosting ISPs offers e-mail, and other web hosting services such as virtual machines, clouds etc.

d) ***Virtual ISPs*****:** Such ISPs offer internet access via other ISP services.

e) ***Free ISPs*****:** Free ISPs do not charge for internet services.

10.6.3 CONNECTION TYPES:

There exist several ways to connect to the internet. Following are these connection types available:

a) Dial-up Connection
b) ISDN
c) DSL
d) Cable TV Internet connections
e) Satellite Internet connections
f) Wireless Internet Connections

a) Dial-up Connection

Dial-up connection uses telephone line to connect PC to the internet. It requires a modem to setup dial-up connection. This modem works as an interface between PC and the telephone line.

There is also a communication program that instructs the modem to make a call to specific number provided by an ISP.Dial-up connection uses either of the following protocols:

1. Serial Line Internet Protocol (SLIP)
2. Point to Point Protocol (PPP)

The following diagram shows the accessing internet using modem:

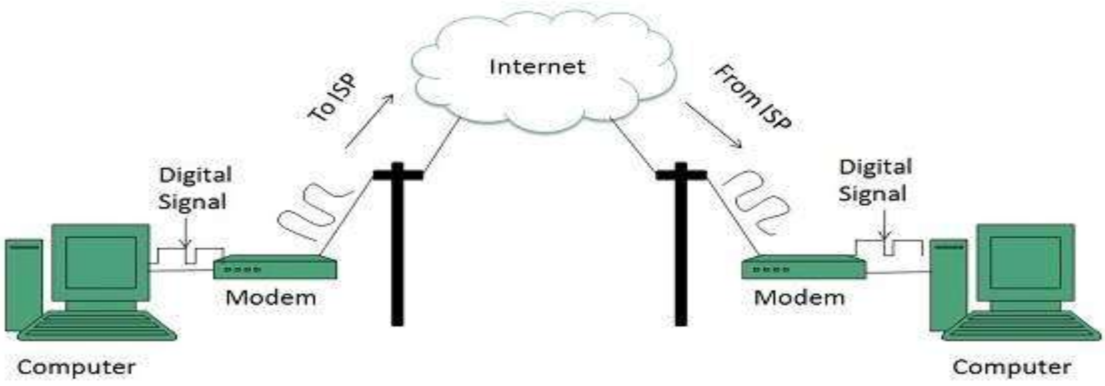


Figure 10.6: Dial-up connection

b) ISDN

ISDN is acronym of ***Integrated Services Digital Network*.** It establishes the connection using the phone lines which carry digital signals instead of analog signals.There are two techniques to deliver ISDN services:

1. **Basic Rate Interface (BRI):** consists of three distinct channels on a single ISDN line: t1o 64kbps B (Bearer) channel and one 16kbps D (Delta or Data) channels.
2. **Primary Rate Interface (PRI):**consists of 23 B channels and one D channels with both have operating capacity of 64kbps individually making a total transmission rate of 1.54Mbps.

The following diagram shows accessing internet using ISDN connection:

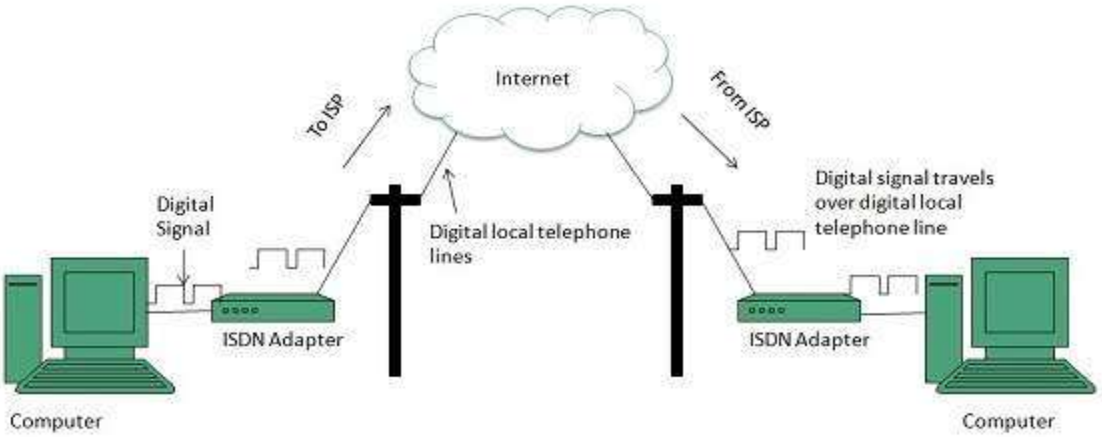


Figure 10.7: ISDN connection

c) DSL

DSL is acronym of **Digital Subscriber Line.** It is a form of broadband connection as it provides connection over ordinary telephone lines.

Following are the several versions of DSL technique available today:

1. Asymmetric DSL (ADSL)
2. Symmetric DSL (SDSL)
3. High bit-rate DSL (HDSL)
4. Rate adaptive DSL (RDSL)
5. Very high bit-rate DSL (VDSL)
6. ISDN DSL (IDSL)

All of the above mentioned technologies differ in their upload and download speed, bit transfer rate and level of service.The following diagram shows that how we can connect to internet using DSL technology:

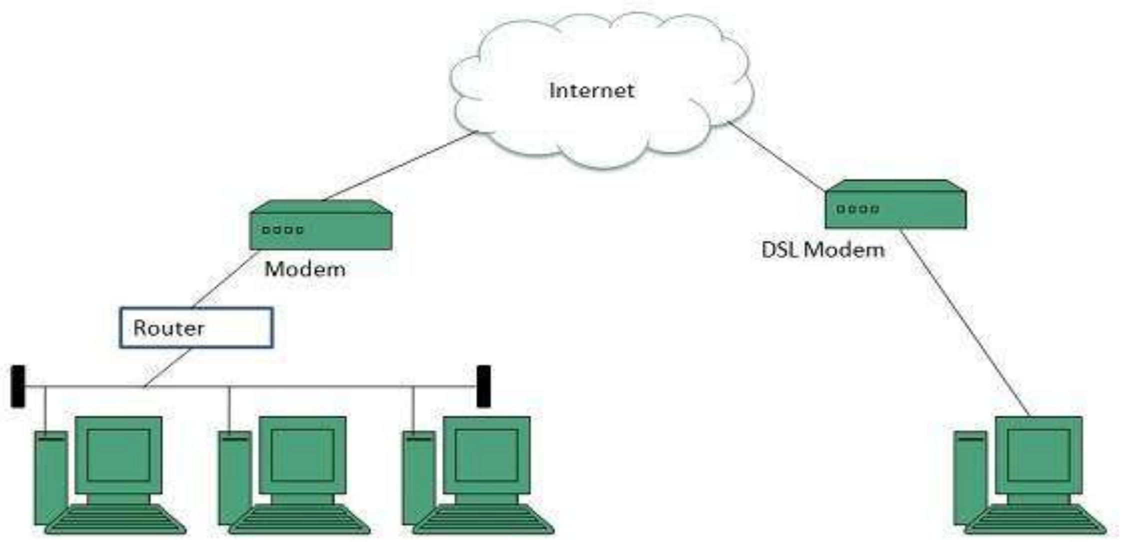


Figure 10.8: Internet Access using DSL

d) Cable TV Internet Connection

Cable TV Internet connection is provided through Cable TV lines. It uses coaxial cable which is capable of transferring data at much higher speed than common telephone line.

A cable modem is used to access this service, provided by the cable operator.The Cable modem comprises of two connections: one for internet service and other for Cable TV signals.Since Cable TV internet connections share a set amount of bandwidth with a group of customers, therefore, data transfer rate also depends on number of customers using the internet at the same time.

The following diagram shows how internet is accessed using Cable TV connection:

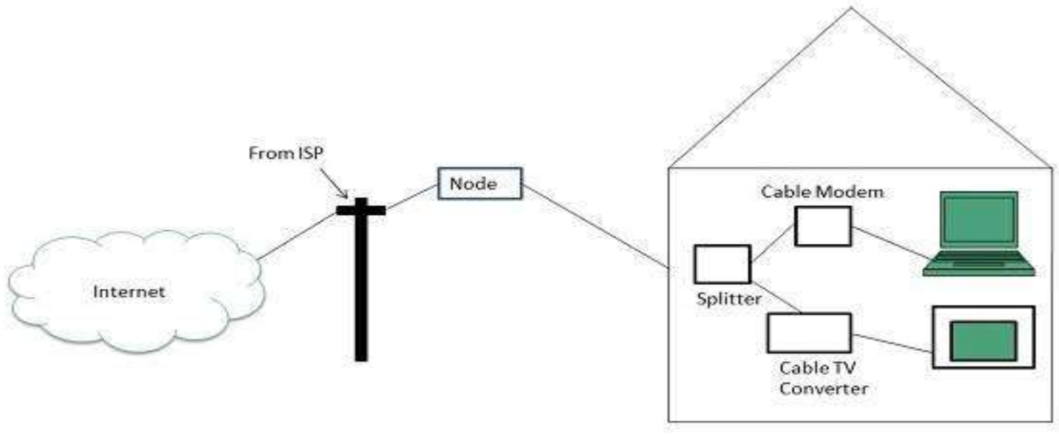


Figure 10.9: Internet access using Cable TV connection

e) Satellite Internet Connection

Satellite Internet connection offers high speed connection to the internet. There are two types of satellite internet connection: one way connection or two way connection.

In *one way connection*, we can only download data but if we want to upload, we need a dialup access through ISP over telephone line.

In *two way connection*, we can download and upload the data by the satellite. It does not require any dialup connection.The following diagram shows how internet is accessed using satellite internet connection:

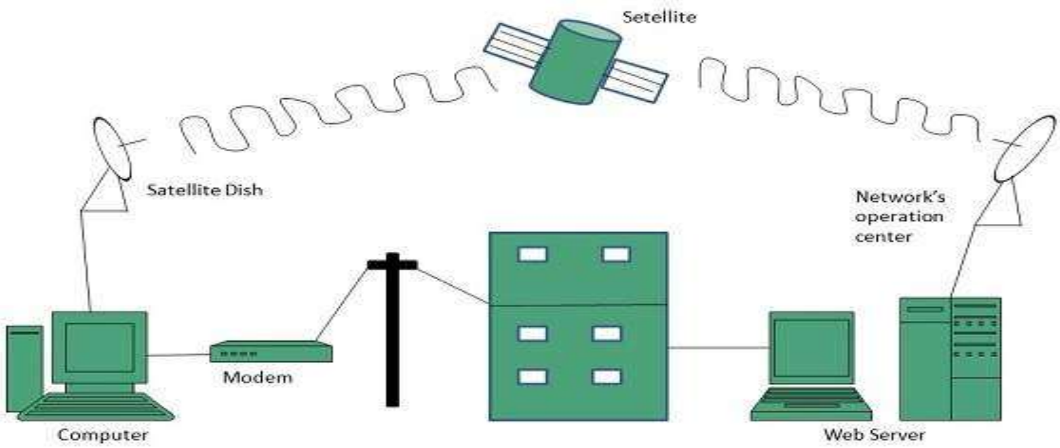


Figure 10.10: Internet access using Satellite connection

f) Wireless Internet Connection

Wireless Internet Connection makes use of radio frequency bands to connect to the internet and offers a very high speed. The wireless internet connection can be obtained by either WiFi or Bluetooth.

Wi Fi wireless technology is based on IEEE 802.11 standards which allow the electronic device to connect to the internet.Bluetooth wireless technology makes use of short-wavelength radio waves and helps to create personal area network (PAN).

10.7 Internet Services

10.7.1 The World Wide Web

The World Wide Web (WWW) is a system on the Internet which allows documents to be connected to other documents by hypertext links, enabling the user to search for information by moving from one document to another. It consists of a large number of web servers that host websites. A website consists of a number of web pages connected by hypertext links. A web page is a text file that contains information stored using a structured language called HTML (Hypertext Markup Language).

A website can be accessed by typing its address or URL (Uniform/Universal Resource Locator) into the address bar of a web browser. An example of a URL is http://www.crtv.cm where http is the protocol used and www.crtv.cm, the domain name (address) of the site.

Example 1: *http://www.gbhsbafoussam.edu/Ls3,4/ict796/intenet.pdf*

- ✓ **http** is the protocol used (hypertext transfer protocol)
- ✓ **www.gbhsbafoussam.edu** is the domain name (*the machine at GBHS bafousssam that hosts the website*)
- ✓ **Ls3,4/ict796/internet.pdf** is the path of the document (resource) on the host computer. Ls3,4 is the folder, ict796 is the subfolder and internet.pdf is the file(resource).

Example 2: *www.minsup.gov.cm*

- ✓ **gov** is the top level domain which specifies that the URL is for a government institution.
- ✓ **cm** specifies the country in which the URL is hosted or the country in which the institution is found.

a) Domain Name System

A domain name system (DNS) is a service which performs the function of turning human-understandable domain names into IP addresses.

b) Web Browser

A web browser (or simply browser) is a computer program that enables a user to read hypertext in files or on the World Wide Web. Popular browsers include Mozilla Firefox, Microsoft Internet Explorer, Opera Mini and Netscape.

c) Search Engine

Search Engine refers to a huge database of internet resources such as web pages, newsgroups, programs, images etc. It helps to locate information on World Wide Web.User can search for any information by passing query in form of keywords or phrase. It then searches for relevant information in its database and return to the user.

Generally there are three basic components of a search engine as listed below:

1. Web Crawler
2. Database
3. Search Interfaces

1) **Web crawler**: It is also known as spider or bots. It is a software component that traverses the web to gather information.

2) **Database**: All the information on the web is stored in database. It consists of huge web resources.

3) **Search Interfaces**: This component is an interface between user and the database. It helps the user to search through the database.

Search Engine Working: Web crawler, database and the search interface are the major component of a search engine that actually makes search engine to work. Search engines make use of Boolean expression AND, OR, NOT to restrict and widen the results of a search. Following are the steps that are performed by the search engine:

- The search engine looks for the keyword in the index for predefined database instead of going directly to the web to search for the keyword.
- It then uses software to search for the information in the database. This software component is known as ***web crawler.***
- Once the web crawler finds the pages, the search engine then shows the relevant web pages as a result. These retrieved web pages generally include title of page, size of text portion, first several sentences etc.

These search criteria may vary from one search engine to the other. The retrieved information is ranked according to various factors such as frequency of keywords, relevancy of information, links etc.User can click on any of the search results to open it.

Some examples of search engines include; Google, Bing, Ask, AltaVista, LYCOS, Alexa, AOL.Search.

10.7.2 Electronic Mail

Electronic mail or e-mail (*email*) is a means of sending messages, text, and computer files between computers via the Internet. To send and receive e-mails, you need an Internet connection and an e-mail account which can be created within a webmail service such as Yahoo, Hotmail or Gmail. When you create an e-mail account, you are given a unique email address that gives you access to your mail box. An email address is made up of two parts separated by the symbol @ pronounced “at”. For example *username@ubea.cm*.
In the above address,

- ✓ ***username*** is the user ID, user name or login
- ✓ ***ubuea.cm*** is the domain name. The domain specifies the mail server (computer) on which the mail box is located.

The part of the domain name after the dot is called top-level domain, and specifies the type of organization or the country the host server is located. Some common top-level domains are:

- ***.com*** - for commercial enterprises
- ***.edu*** - for educational institutions and universities
- ***.gov*** - for United States government agencies
- ***.net*** - for organizations such as Internet Service Providers
- ***.org*** - for non-commercial organizations

10.7.3 Instant Messaging

Instant messaging is a live (or real time) communication which occurs when brief text messages are exchanged instantly over the Internet. Instant Messaging requires that both users be on-line at the same time. Common IM applications are AOL Instant Messenger, Yahoo Messenger and Microsoft MSN messaging.

10.7.4 Internet Telephony

Internet telephony or voice over IP (VoIP) is the transmission of voice telephone conversations through the Internet or IP networks. It allows users to have voice-talk with others. The telephone calls are digitized and transmitted through the Internet. Internet telephone services can be mainly categorized into net-to-net and net-to-phone telephony.

In net-to-net telephony, both caller and receiver must be online. When both are online, one dials the other person's phone number. If they accept the call, then voice communication is established.

In net-to-phone, only one person has to be online. This person dials the other person's phone number and the latter receives a ring on their phone. Yahoo messenger and Skype provide services for both types.

10.7.5 Interpersonal Computing

Interpersonal computing refers to person-to-person interactions facilitated by websites that enable collaborative content creation, sharing and manipulation. Interpersonal computing involves: blogs, social networks, wikis and viral video sites.

a. Blogs

A blog (web log) is a chronological, journal-style website which its author (or "blogger") maintains like an online diary, with regular entries of commentary, descriptions of events, or other material such as graphics or video. Many blogs provide commentary or news on a particular subject; others function as more personal online diaries. They also provide the readers with the ability to leave comments in an interactive format.

b. Social Networking Sites

Social networking sites are websites that allow user to build personalized communities to socialize with. Common features include a customizable profile, the ability to add other users as friends, the ease of sharing pictures, music, text, and links, and built-in chat and mail features. Examples of social networking sites are Facebook, Twitter and Instagram (*see section 10.9 for more information*).

c. Wikis

Wikis are websites that allow visitors to easily add, remove and edit content, hence enabling the collaborative authorship of comprehensive documents. The best example of a wiki is the multi-lingual, web-based encyclopediaWikipedia, and which currently includes over two million articles.

d. Viral Video Sites

A viral video is a video that is distributed by sharing. Viral video sites are websites that allow anybody to post videos online. Whilst it is now not difficult to put a video on any website, the significance of viral video sites is that they provide somewhere to put videos where it is likely that at least some other people will actually find them. Examples are YouTube and Kaltura.

10.7.6 Electronic Commerce

E-commerce refers to the buying and selling on the Internet. Different models of e-commerce exists: business-to-business, business-to-consumer, business-to-government and m-commerce

a) Business-to-Consumer

B2C model sells goods or services to the consumer, generally using online catalog and shopping cart transaction systems. For example, an online pharmacy giving free medical consultation and selling medicines to patients is following B2C model. Amazon is an example of one of the first and still one of the most successful B2C e-commerce companies.

b) Business-to-Business

B2B describes commerce transactions between businesses, such as between a manufacturer and a wholesaler, or between a wholesaler and a retailer. In this form, the buyers and sellers are both business entities and do not involve an individual consumer.

c) Business-to-Government

B2G is a derivative of B2B marketing. B2G sites provide a platform for businesses to bid on government opportunities which are presented as solicitations requests for proposal (RFPs) to tender.

d) M-Commerce

M-commerce refers to the use of mobile devices for conducting transactions. The mobile device holders can contact each other and can conduct the business. Even the web design and development companies optimize the websites to be viewed correctly on mobile devices.

Some e-commerce websites are: *www.bruneiair.com* for airline ticket bookings, *www.amazon.com* for sales of books and magazines, *www.jumia.com/shop* for sales of computers, shirts, and cameras

i. Some *advantages of setting up an e-commerce website* are:
   - Products can be sold to local customers and those from abroad.
   - It is accessible 24 hours each day.
   - It needs a small number of staff to run.

- It does not need huge office space.
- Products can be sold at cheap prices

ii. Some *disadvantages of e-commerce are*:

- Credit card fraud - hackers are able to steal credit card numbers on computers.
- Certain websites spy or track the buying habits of their customers.
- Some goods do not arrive after they are paid for.
- It lacks human interaction as one only sees pictures and some text descriptions.

10.7.7 Online Banking

Online banking (Internet banking) is simply the use of the Internet to perform banking operations like opening an account, accessing account information, transferring funds, getting a bank statement etc. In an Internet banking system, the bank has a centralized database that is web-enabled. All the services that the bank has permitted on the Internet are displayed in a menu. Any service can be selected and further interaction is dictated by the nature of service.

10.8 Protocols

10.8.1 Internet Protocol

Internet Protocol (IP) specifies the format of packets and the addressing scheme. All computer devices (desktops, laptops, PDAs, phones, tablets) connected to the Internet, have IP addresses by which they are identified.

Definition*: An IP address is a unique identifying number given to every single computer on a TCP/IP network.*

Two versions of IP addresses are available: **IPv4** that uses 32 bits and **IPv6** that uses 128 bits.

a) An IPv4 is made up of four sets of numbers separated by dots such as 123.23.168.22. This notation is known as dotted decimal notation. Each of the four numbers separated by dots can be any number from 0 to 255, making for a total of 4.3 billion potential IPv4 addresses (i.e. 255×255×255×255).

b) An IPv6 has eight sets of numbers separated by colons such as 3ffe:1900:4545:3:200:f8ff:fe21:67cf.

IP addresses are assigned manually (by an administrator) or automatically (by DHCP or APIPA). An IP address is also known as a ***logical address***.

a) Address Resolution Protocol: ARP resolves IP addresses to MAC addresses.

b) Internet Control Message Protocol: ICMP is responsible for diagnostics and error reporting.

c) Internet Group Management Protocol: IGMP is responsible for management of group multicast.

10.8.1.1 Transmission Control Protocol (TCP)

TCP is a connection-oriented reliable protocol used in the accurate transmission of large amounts of data. Data packets are verified using checksums and retransmitted if they are missing or corrupted. The application plays no part in validating the transfer.

10.8.1.2 User Datagram Protocol (UDP)

UDP is a connectionless unreliable protocol used for the transmission of small amounts of data. Data packets are sent without testing to verify whether they actually arrive at the destination, nor whether they were corrupted in transit. It is up to the application to determine these factors and request retransmissions. UDP is faster compared to TCP.

10.8.1.3 Hypertext Transfer Protocol (HTTP)

HTTP is a standard method of publishing information as hypertext in HTML format on the Internet. It provides the ability to supply web pages between a browser and the server. HTTPS is a secure version of HTTP used for accessing secure web servers, whereby all data transferred are encrypted.

10.8.1.4 File Transfer Protocol (FTP)

FTP is a standard for transferring files between a server and a client on a TCP/IP network. It provides the ability to upload and download files between hosts on the network.

10.8.1.5 Simple Mail Transfer Protocol (SMTP)

SMTP is used for sending e-mails between servers on the Internet and other TCP/IP networks. It governs the transmission of mail messages and attachments. SMTP is used in the case of outgoing messages.

10.8.1.6 Post Office Protocol (POP)

POP is a standard protocol for delivering e-mails to personal computers. There are different versions of the post office protocol indicated by POP*n*where *n = 1, 2, 3* or*4.*

10.8.1.7 Telnet

Telnet is a protocol that allows a computer on the network to be accessed remotely. It provides the ability to login into a remote host and administer the machine. Using Telnet a computer can be used as a terminal on another.

10.8.1.8 Wireless Application Protocol (WAP)

WAP is a protocol which runs on mobile phones and provides a universal open standard for bringing Internet content to mobile phones and other wireless devices.

10.9 TCP/IP Ports

A computer has a single physical connection to the network. All data destined for a particular computer arrives through that connection. However, the data may be intended for different applications running on the computer. To identify the application for which the data is intended, TCP requires port numbers on the host and destination for communication.
***Definition**: A communication port is a 16-bit number that identifies an application on the Internet or TCP/IP network.*

Popular Internet application protocols are associated with well-known ports assigned by the Internet Assigned Number Authority (IANA). Sample TCP port numbers are:

Port number	Protocol
20	FTP data channel
21	FTP control channel
23	Telnet
25	SMTP
80	HTTP
110	POP

Table 10.3: *TCP Port Numbers and their protocols*

Ports are usually combined with IP addresses to form a socket. For example *127.102.10.0:80.*

10.10 Social Networks

Social Networking refers to grouping of individuals and organizations together via some medium, in order to share thoughts, interests, and activities.

There are several web based social network services available such as facebook, twitter, linkedin, Google+ etc. which offer easy to use and interactive interface to connect with people with in the country and overseas as well. There are also several mobile based social networking services in the form of apps such as Whatsapp, hike, Line etc.

10.10.1 Available Social networking Services

The following table describes some of the famous social networking services provided over web and mobile:

S.N.	Service Description
1.	**Facebook** Allows to share text, photos, video etc. It also offers interesting online games.
2.	**Google+** It is pronounced as Google Plus. It is owned and operated by Google.
3.	**Twitter** Twitter allows the user to send and reply messages in form of tweets. These tweets are the small messages, generally include 140+ characters.

4.	**Faceparty** Faceparty is a UK based social networking site. It allows the users to create profiles and interact with each other using forums messages.
5.	**Linkedin** Linkedin is a business and professional networking site.
6.	**Flickr** Flickr offers image hosting and video hosting.
7.	**Ibibo** Ibibo is a talent based social networking site. It allows the users to promote one's self and also discover new talent.
8.	**Whatsapp** It is a mobile based messaging app. It allows to send text, video, and audio messages
9.	**Line** It is same as whatsapp. Allows to make free calls and messages.
10.	**Hike** It is also mobile based messenger. Allows to send messages and exciting emoticons.

Table 10.4: *Social networking services*

10.10.2 Applications of Social Networks:

Following are the areas where social networking has become most popular:

a) ***Online Marketing***: Websites like facebook allows us to create a page for specific product, community or firm and promoting it over the web.

b) ***Online Jobs***: Websites like linkedin allows us to create connection with professionals and helps to find the suitable job based on one's specific skills set.

c) ***Online News***: On social networking sites, people also post daily news which helps to keep us updated.

d) ***Chatting*****:** Social networking allows us to keep in contact with friends and family. We can communicate with them via messages.

e) ***Share Picture, Audio and video***: One can share picture, audio and video using social networking sites.

10.10.3 Advantages of Social Networks

1. ***Staying Connected***
The main purpose of social media is to be able to stay connected to friends and families in today's fast paced and ever changing worlds. You are able to rekindle old friendships, share family photos, and special events in your life with just about everyone you know, at the same time.

2. ***Finding People with Common Interests***
Social networking is also a great way to meet entirely new people. You can seek out groups that are focused towards your special interests and hobbies and connect with local people that share the same interests. Online and social media dating is almost more common than traditional dating is in today's world.

3. ***Invaluable Promotional Tool***
Companies, artists, and musicians can reach an impossibly large and diverse amount of people using social media sites. This allows them to promote and market themselves and their products in a way that has never been seen before.

4. ***Information Spreads Incredibly Fast***
Breaking news and other important information can spread like wildfire on social media sites. Important things like recalls, storm information, or missing children are all communicated and taken seriously very quickly.

5. ***Helps to catch and convict Criminals***
People often do not think of the consequences of what they post of these social sites. Pictures of themselves doing illegal things, or even bragging posts about crimes they have committed are all things that law enforcement use to persecute these criminals. They also use these sites to identify and solve existing cases.

10.10.4 Disadvantages of Social Networks

1. ***Perpetuates False and Unreliable Information***
Just like stated above, anything can spread to millions of people within hours or days on social media. This also, unfortunately, includes things that are false or made up. This information can cause panic and severe misinformation in society.

2. ***Causing Major Relationship Problems***
Online social interactions with social networking have not only been starting new relationships, but ending many others. It is very simple to communicate and share pictures or plans with a person on social media and keep it completely under wraps. This new temptation has been driving wedges into people's real life, offline relationships, often time ending them for good. Social networking puts trust to the limit.

3. ***Cyber Bullying Is A Growing Problem***
Having access to people's lives at all times is not always a good thing. A new trend of cyber

bullying is wreaking havoc all across the world. This is especially true with young kids. They are publicly harassing one another, and posting mean or slanderous things which are broadcasted to the entire cyber world.

4. *Used To Profile and Discriminate in the Job World*
Just about everyone has a social media account the shows what they look like, the type of life that they live, and how old they are. Employers are using this to their advantages is some very unsettling ways. Jobs that are looking for a certain criteria of person, but cannot legally express these criteria, are using social media to prescreen their applicants.

5. *The Addiction Is Real*
One of the biggest problems with the social media craze is that people are becoming more and more addicted to using it. It is the number one time waster at work, in school, and at home. All of this has caused people to have literal withdraws from their social networks.

10.11 Web Design with HTML

HTML stands for **H**yper**T**ext **M**arkup **L**anguage. It is the authoring language that describes how a Web page should be displayed by a Web browser. ***Hypertext*** means that it provides ways of representing information with links or connections to other information. These links are called *hypertext links*. ***Markup*** means that it provides ways to indicate underlining, italics, paragraph breaks, section headings, and so on, in text.

10.11.1 HTML Documents

An HTML document is simply a text file that is saved with the extension *.html* or *.htm*. It can be created by a simple text editor like *Microsoft Notepad*, *Notepad++* as well as a sophisticated web authoring tool like *FrontPage* or *Dreamweaver*.

HTML documents are made up of mark-up tags (or simply tags). A tag is a code enclosed within angle brackets that indicates how something is to be interpreted by a web browser. Some examples of tags are <html>,
 and <em>. Most tags come in pairs: an open tag and a closing tag which is written with a slash after the first < (e.g. </html>). A closing tag tells the browser where to stop applying the effect of a given tag.

An HTML document is contained within the <HTML> and </HTML> tags and comprises of two sections: **head** and **body**.

i) The **Head** contains the page title and meta-tags within the <HEAD></HEAD> tags. Any JavaScript code that is used, as well as Cascading Style Sheet information is also contained within the Head. This section will not be displayed on the web page.

ii) The **Body** holds the actual content of the page (text, graphics, lists, etc.) contained within the <BODY></BODY> tags.

The <HTML>, <HEAD>, <TITLE>, and <BODY> tags are referred to as document tags while the tags that are used within the body part of the document are known as markup tags.
A basic HTML document would look like this:

```
<HTML>
   <HEAD>
        <TITLE>Page title here</TITLE>
   </HEAD>
  <BODY>
        Page content here.
  </BODY>
</HTML>
```

- ✓ **<HTML>** marks the beginning of an HTML document
- ✓ **<HEAD>** begins the heading section of the document
- ✓ **<TITLE> ... </TITLE>** gives a title that will appear on the browser's menu bar. This section must appear between the <HEAD> ... </HEAD> tags and should be straight text, no tags.
- ✓ **</HEAD>** defines the end of the heading
- ✓ **<BODY> … </BODY>** defines the body of the document (text contained within the <BODY> … </BODY> tags appears in the main browser window). It can be used with the attribute BGCOLOR.
- ✓ **</HTML>** defines the end of the document

Remark*Once an HTML document has been saved, it can now be opened as a web page using a web browser.*To edit the page, go back to the text document. Make the changes and save the document. In the web page, click "reload" or "refresh" to apply the changes.

10.11.2 Text Tags

Text tags are used to format text within the document. Some tags are used with attributes. An attribute is a special code that can enhance or modify a tag. They are generally located in the starting tag after the tag name. The basic syntax for html tags and attributes is:

<tag attribute="value"> … </tag>

Tags	Description
 … or … 	Bolds the text between the opening and closing tags
<i>...</i> or ...	Puts text in italics
<u> … </u>	Underlines text
 … … 	Sets the appearance of the text in your page. Can be used with "size", "color" and "face" attributes.
<blink> … </blink>	Causes text to blink
<center> … </center>	Centers text or any item or group of items place between its open and closing tags
<marquee> … </marquee>	Causes text to navigate

… …	Sets a paragraph apart from other text and adds a line break after. </p> is optional
<!--...-->	Used to insert comments. Comments are not displayed by the browser. They are footnotes for you.
 	Used to insert a line break. It does not have a closing tag.

Table 10.5: *Some HTML tags and their descriptions*

Example 1:

```
<html>
<head>
<title>My First Web Page</title>
</head>
<body>
        <strong>This text is bold</strong><br>
        <b>This text is bold too</b><br>
        <em>This text is in italics</em><br>
        <i>This text is in italics too</i><br>
        <u>This text is underlined</u><br>
        <b><i>This text is bold and in italics</i></b><br>
        <strong><em><u>This text is bold, underlined and in
        italics</u></em></strong>
</body>
</html>
```

Example 2:

```
<html>
<head>
<title>MyPage</title>
</head>
<bodybgcolor="cyan">
        <font type="Monotype corsiva" size= "12" color="blue"
</body>
</html>
```

Remark *Tags are case insensitive but it is advisable to write them in lower case.All attribute values must be written within double quotes.*

10.11.3 Heading Tags

Heading tags are used to highlight text by making them bigger and bolder than normal text. There are six levels of headings numbered **1** through **6**, where 1 is the biggest and 6 is the

smallest heading. Headings are specified as <hy></hy> where h stands for heading and y is the level number (1 to 6).

Example:

```
<html>
<head>
        <title>Heading</title>
</head>
<body>
        <h1>Level 1 heading</h1>
        <h2>Level 2 heading</h2>
</body>
</html>
```

10.11.4 List Tags

List tags are used to display items that are arranged in a list. Two types of lists exist: ordered and unordered lists.
Their respective tags are <ol> … and <ul> and . No matter the type of list, the <li> … tag is used for every item inside the list.

Example:

```
<ol>                <ul>
<li>item 1</li>     <li>item 1</li>
<li>item 2</li>     <li>item 2</li>
<li>item 3</li>     <li>item 3</li>
</ol>               </ul>
```

10.11.5 Anchor (Link) Tag

A link can be text or an image which when clicked takes one to another part of the document, another html document on the same machine or to another website. A link is introduced using the anchor <a> ... tag.

```
<a href="URL or address"> … </a>
<a href="www.facebook.com">Click here to log onto to Facebook</a>
<a href="mailto:joshnoyah@ymail.com">Send a mail to Josh</a>
<a href="#top">Back to top of page </a>
```

10.11.6 Image Tag

Images are inserted in the page with the use of the image tag <img>. It has no closing tag and can be used with the attributes src which specifies the source of the image, align, top and width.

Example:

```
<imgsrc= "image path" align="top" height="40" width="30"> … </img>
<imgsrc="logo.jpg">
<imgsrc= "mypict.gif>
```

10.11.7 Table Tags

Tables are defined with the <table> tag. A table is divided into rows with the <tr> tag, and each row is divided into data cells with the <td> tag. The letters td stands for table data, which is the content of a data cell. A data cell can contain text, images, lists, paragraphs, forms, horizontal rules, tables, etc.

Example:

```
<table>
<tr><td>row 1, cell 1</td>
<td>row 1, cell 2</td>
</tr>
<tr><td>row 2, cell 1</td>
<td>row 2, cell 2</td>
</tr>
</table>
```

10.11.8 Form tags

An HTML form is a set of one or more input controls and a submit button. The form controls are surrounded by a `<form>`tag which has two important attributes: **action** and **method**. Action is the URL destination where the information will be sent and Method defines how the information will be sent (either "post" or "get").

Forms themselves are not visible but contain visible elements such as list-boxes and drop-down lists.

Example:

```
 <form action="http://someurl.com" method="get">
 <input name="first" value="Hi">
 <input type="submit" value="Send">
</form>
```

The example above is just an overview of how the form tag works. Below are some form fields with their tags.

a) **Radio Buttons** - options selection where the user can select only one of several options

```
<form>
<input type="radio" name="choice" value="A">Choice A<br>
<input type="radio" name="choice" value="B">Choice B
</form>
```

b) **Check Boxes** - the user can select multiple options:

```
<form>
  Subscribe <input type="checkbox" name="subscribe"><br>
  Send Samples  <input type="checkbox" name="samples" checked>
</form>
```

c) **Text Field** - a text input box like in the example above

```
<form>
<input type="text" name="value" value="TypeHere">
</form>
```

d) Drop-down Box - a text input box like in the example above

```
<form> Month:
<select name="Month">
<option value="1"> January
<option value="2"> February
<option value="3"> March
<option value="4"> April
</select>
</form>
```

e) Hidden - this type does not show at all! It is used to pass information on to the server which you do not want the user to either see or edit.

```
<form>
<input type="hidden" name="page" value="page 12">
</form>
```

Drill Questions

1. What is a home page?

Answer*: A web page that links a user to the other areas of the website.*

2. Differentiate between an IP address and a MAC address.
3. Give the full meaning of the following protocols and state their functions.
i) IMAP ii) RIP iii) DNS

4. What is meant by GPS? Give two examples of the use of GPS.

5. A desktop computer is already used in an office which does not have an internet connection.
i) Name and explain two items of hardware which would enable the computer to be connected to a Local Area Network
ii) Name and explain one item of hardware which would enable the network to be connected to the internet. (**CGCEB**)

6. Social Network has changed the behaviour of many youths in the society.
i) What is a social network service?
ii) Describe two features of a social network site.
iii) Give three advantages and disadvantages of social networks to the society. (**CGCEB**)

7.i) What is internetworking? Define the term *The Internet.*
ii) Describe the terms: Intranet, Extranet, DNS, Email, and Browser
iii) What is a search Engine? State one use of search engines.
(iv)(a) Whats the difference between internet and extranet?
(c) Explain the meaning of the term netiquette. State ONE important rule that can serve as a guideline for cyberspace behaviour. (**CSC Q2, CGCE 2013**)

8.(a) Explain the consequence of network communication without each of the following protocols (relate each answer to the purpose of each protocol)
a) HTTP b) FTP c) Telnet d) TCP/IP
(b) state two wireless standards and give an estinamtion of their area of coverage (**Q4iii, CGCE 2013**)

CHAPTER 11: ELECTRONIC SERVICES

11.1 E – COMMERCE

11.1.1 Definition and activities involved in E-commerce

Electronic commerce or e-commerce refers to a wide range of online business activities for products and services. It also pertains to "any form of business transaction in which the parties interact electronically rather than by physical exchanges or direct physical contact."

E-commerce is usually associated with buying and selling over the Internet, or conducting any transaction involving the transfer of ownership or rights to use goods or services through a computer-mediated network. Though popular, this definition is not comprehensive enough to capture recent developments in this new and revolutionary business phenomenon. A more complete definition is:

E-commerce is the use of electronic communications and digital information processing technology in business transactions to create, transform, and redefine relationships for value creation between or among organizations, and between organizations and individuals.

11.1.2Advantages and disadvantages of E-commerce

a) The Advantages

1. ***Cost Effective***: The entire financial transactions will eventually become electronic, so sooner conversion is going to be lower on cost. It makes every transaction through e-commerce payment a lot cheaper.

2. ***Higher Margin***: E-commerce also enables us to move better with higher margin for more business safety. Higher margin also means business with more control as well as flexibility. You can also save time from the e-commerce.

3. ***Better Productivity***: Productivity here means productivity for both companies and customers. People like to find answers online because it is faster and cheaper, and it costs a lot cheaper expense as well for the company.

4. ***Quick Comparison***: E-commerce also enables you to compare price among several providers. In the end, it leads you to smart shopping. People can save more money while they shop.

5. ***Economy Benefit***: E-commerce allows us to make transaction without any needs on stores, infrastructure investment, and other common things we find. Companies only need well built website and customer service.

b) The Disadvantages

1. ***Security***: Customers need to be confident and trust the provider of payment method. Sometimes, we can be tricked. Examine on integrity and reputation of the web stores before you decide to buy.

2. ***Scalability of System***: A company definitely needs a well developed website to support numbers of customers at a time. If your web destination is not well enough, you better forget it.

3. ***Integrity on Data and System***: Customers need secure access all the time. In addition to it, protection to data is also essential. Unless the transaction can provide it, we should refuse for e-commerce.

4. ***Products People***: People who prefer and focus on product will not buy online. They will want to feel, try, and sit on their new couch and bed.

5. ***Customer Service and Relation Problem***: They sometimes forget how essential to build loyal relationship with customers. Without loyalty from customers, they will not survive the business.

11.2 E-BANKING

11.2.1 Definition

Online banking is *an electronic payment system that enables customers of a financial institution to conduct financial transactions on a website operated by the institution, such as a retail bank, virtual bank, credit union or building society*. Online banking is also referred as *internet banking*, *e-banking*, *virtual banking* and by other terms.

To access a financial institution's online banking facility, a customer with Internet access would need to register with the institution for the service, and set up a password and other credentials for customer verification. The credentials for online banking is normally not the same as for telephone banking. Financial institutions now routinely allocate customers numbers, whether or not customers have indicated an intention to access their online banking facility. Customers' numbers are normally not the same as account numbers, because a number of customer accounts can be linked to the one customer number. The customer number can be linked to any account that the customer controls, such as cheque, savings, loan, credit card and other accounts.

To access online banking, a customer visits the financial institution's secure website, and enters the online banking facility using the customer number and credentials previously setup. Online banking services usually include viewing and downloading balances and statements, and may include the ability to initiate payments, transfers and other transactions, as well as interacting with the bank in other ways.

11.2.2Advantages and disadvantages

a) Advantages

1. It's generally secure. But make sure that the website you're using has a valid security certificate. This let you know that the site is protected from cyber-thieves looking to steal your personal and financial information.
2. You have twenty-four-hour access. When your neighborhood *bank* closes, you can still access your account and make transactions online. It's a very convenient alternative for those that can't get to the bank during normal hours because of their work schedule, health or any other reason.
3. You can access your account from virtually anywhere. If you're on a business trip or vacationing away from home, you can still keep a watchful on your money and financial transactions - regardless of your location.
4. Conducting business online is generally faster than going to the bank. Long teller lines can be time-consuming, especially on a Pay Day. But online, there are no lines to contend with. You can access your account instantly and at your leisure.
5. Many features and services are typically available online. For example, with just a few clicks you can apply for *loans*, check the progress of your *investments*, review *interest rates* and gather other important information that may be spread out over several different brochures in the local bank.

b) Disadvantages

1. Online banking is generally secure, but it certainly isn't *always* secure. *Identity theft* is running rampant, and banks are by no means immune. And once your information is compromised, it can take months or even years to correct the damage, not to mention possibly costing you thousands of dollars, as well.
2. Some online banks are more stable than others. Not all online setups are an extension of a brick-and-mortar bank. Some operate completely in cyberspace, without the benefit of an branch that you can actually visit if need be. With no way to physically check out the operation, you must be sure to thoroughly do your homework about the bank's background before giving them any of your money.
3. Before using a banking site that you aren't familiar with, check to make sure that their deposits are **FDIC** (*Federal Deposit Insurance Cooperation*) - insured. If not, you could possibly lose all of your deposits if the bank goes under, or its major shareholders decide to take an extended vacation in "*Switzerland*".
4. Customer service can be below the quality that you're used to. Some people simply take comfort in being able to talk to another human being face-to-face if they experience a problem. Although most major banks employ a dedicated customer service department specifically for online users, going through the dreaded telephone menu can still be quite irritating to many. Again, some are considerably better (or worse) than others.
5. Not all online transactions are immediate. Online banking is subject to the same business-day parameters as traditional banking. Therefore, printing out and keeping receipts is still very important, even when banking online.

11.3E-HEALTH

11.3.1 Definition

E-health is the transfer of *health* resources and *health* care by electronic means. It encompasses three main areas: The delivery of *health* information, for *health* professionals and *health* consumers, through the Internet and telecommunications. Below are points which charaterises what E-health should be.

a) ***Efficiency*** - one of the promises of e-health is to increase efficiency in health care, thereby decreasing costs. One possible way of decreasing costs would be by avoiding duplicative or unnecessary diagnostic or therapeutic interventions, through enhanced communication possibilities between health care establishments, and through patient involvement.
b) ***Enhancing quality*** of care - increasing efficiency involves not only reducing costs, but at the same time improving quality. E-health may enhance the quality of health care for example by allowing comparisons between different providers, involving consumers as additional power for quality assurance, and directing patient streams to the best quality providers.
c) ***Evidence based*** - e-health interventions should be evidence-based in a sense that their effectiveness and efficiency should not be assumed but proven by rigorous scientific evaluation. Much work still has to be done in this area.
d) ***Empowerment*** of consumers and patients - by making the knowledge bases of medicine and personal electronic records accessible to consumers over the Internet, e-health opens new avenues for patient-centered medicine, and enables evidence-based patient choice.
e) ***Encouragement*** of a new relationship between the patient and health professional, towards a true partnership, where decisions are made in a shared manner.
f) ***Education*** of physicians through online sources (continuing medical education) and consumers (health education, tailored preventive information for consumers)
g) ***Enabling*** information exchange and communication in a standardized way between health care establishments.
h) ***Extending*** the scope of health care beyond its conventional boundaries. This is meant in both a geographical sense as well as in a conceptual sense. E-health enables consumers to easily obtain health services online from global providers. These services can range from simple advice to more complex interventions or products such a pharmaceuticals.
i) ***Ethics*** - e-health involves new forms of patient-physician interaction and poses new challenges and threats to ethical issues such as online professional practice, informed consent, privacy and equity issues.
j) ***Equity*** - to make health care more equitable is one of the promises of e-health, but at the same time there is a considerable threat that e-health may deepen the gap between the "haves" and "have-nots". People, who do not have the money, skills, and access to computers and networks, cannot use computers effectively. As a result, these patient populations (which would actually benefit the most from health information) are those who are the least likely to benefit from advances in information technology, unless political measures ensure equitable access for all. The digital divide currently runs between rural vs. urban populations, rich vs. poor, young vs. old, male vs. female people, and between neglected/rare vs. common diseases.

11.3.2 Advantages and Disadvantages

a) Advantages:

1) Reduces staff stress.
2) Makes an efficient and accessible patient record.
3) Time saving and reducing indirect works, that leads to more direct care delivery.
4) Causes keeping staff in their possession and attracting them.
5) Provides anonymity to users

b) Disadvantages:

1) Concerns regarding treatment credibility
2) User privacy and confidentiality.
3) Changes communication processes with staff.
4) Training of staff is quite time-consuming.

11.4 E-GOVERNMENT

11.4.1 Definition

"E-Government" refers to the use by government agencies of information technologies (such as Wide Area Networks, the Internet, and mobile computing) that have the ability to transform relations with citizens, businesses, and other arms of government.

These technologies can serve a variety of different ends: better delivery of government services to citizens, improved interactions with business and industry, citizen empowerment through access to information, or more efficient government management. The resulting benefits can be less corruption, increased transparency, greater convenience, revenue growth, and/or cost reductions.

Traditionally, the interaction between a citizen or business and a government agency took place in a government office. With emerging information and communication technologies it is possible to locate service centers closer to the clients. Such centers may consist of an unattended kiosk in the government agency, a service kiosk located close to the client, or the use of a personal computer in the home or office.

Analogous to e-commerce, which allows businesses to transact with each other more efficiently (B2B) and brings customers closer to businesses (**B2C**), e-government aims to make the interaction between government and citizens (**G2C**), government and business enterprises (G2B), and inter-agency relationships (**G2G**) more friendly, convenient, transparent, and inexpensive.

11.4.2 Advantages and Disadvantages

a) The Advantages of Electronic-Government

The ultimate goal of the e-government is to be able to offer an increased portfolio of public services to citizens in an efficient and cost effective manner. It allows for government transparency because it allows the public to be informed about what the government is working on as well as the policies they are trying to implement.

The main advantage while implementing electronic government will be to improve the efficiency of the current system (*Paper based system*). That would in return same money and time. The introduction would also facilitate better communications between governments and businesses.
For example: E-procurement facilitates ***G2G*** and ***B2B*** communication; this will permit smaller business to compete for government contracts as well as larger business. Hence the advantage of creating an open market and stronger economy. Business and citizens can obtain information at a faster speed and it is possible at any time of the day.

The society is moving towards the mobile connections and the ability of an e-government service to be accessible to citizens irrespective of location throughout the country brings the next and potentially biggest benefit of an e-government service as we live in what is now termed as the Knowledge era.

The anticipated benefits of e-government include efficiency, improved services, better accessibility of public services, and more transparency and accountability.

b) The Disadvantages of Electronic-Government
The main disadvantage concerning e-government is the lack of equality in public access to the internet, reliability of information on the web, and hidden agendas of government groups that could influence and bias public opinions.
Potential implications of implementing and designing e-government, include disinter-mediation of the government and citizens, impacts on economic, social and political factors, vulnerability to cyber-attacks and disturbance to the status quo in these areas.

Hyper-surveillance: Once the government begins to develop and become more sophisticated, the citizens will be forced to interact electronically with the government on a larger scale. E.gThis could potentially lead to a lack of privacy for civilians as their government obtains more and more information on them.

Cost: Although large amount of money is spent on the development and implementation of e-government the outcomes and effects of trial internet-based governments are often difficult to gauge or unsatisfactory.

Inaccessibility: An e-government site that provides web based access and support often does not offer the potential to reach many users including those who live in the remote areas, have low literacy levels and exist on poverty line incomes.

11.5 COMPUTER ASSISTED LEARNING

11.5.1 Definition:

The term Computer Assisted Learning (CAL) covers a range of computer-based packages, which aim to provide interactive instruction usually in a specific subject area, and many predate the Internet. These can range from sophisticated and expensive commercial packages to applications developed by projects in other educational institutions or national initiatives to simple solutions developed by individuals with no funding or support to tackle a very local problem. The amount of time and money invested in development is high and partly because of the very subject specific nature of the education market as well as the very personalised nature of the teaching process - particularly at Primary Education and Secondary Education level - means that commercial success is difficult to achieve and work done in one subject area rarely transfers to others subject areas.

In general, the use of computers in education through CAL has been sporadic a great deal of effort was expended with little general impact. Many of those academics that took part in that earlier crusade are now cynical about the effectiveness of computers in teaching.

There are still good reasons to use CAL rather than Internet based technologies. CAL is run either straight from a CD or floppy disk drive, or over a local network so the constraint of the internet - slow download times for multimedia materials may not apply. This, coupled with the fact that CAL technology has been around a bit longer, means that CAL packages have the potential to offer more advanced, interactive, multimedia learning experiences than it is currently reasonable to expect from the Web. This has been changing as Web technologies develop and bandwidths improve but there are currently many things that can only be achieved with CAL rather than the Web and CAL has been an integral part of the curriculum in many departments at Warwick for some time

11.5.2 Advantages and Limitations of Computer assisted Learning

a) Advantages

1. CAL is individualized, that is each student is free to work at his own place, totally unaffected by the performance of any other students.

2. Information is presented in a structured form. It proves useful in the study of a subject where there is hierarchy of facts and rules.

3. CAL forces active participation on the part of the student, which contrasts with the more passive role in reading a book or attending a lecture.

4. CAL utilizes a reporting system that provides the student with a clear picture of his progress. Thus students can identify the subject areas in which they have improved and in which they need improvement.

5. By enabling students to manipulate concepts directly and explore the results of such manipulation, it reduces the time taken to comprehend difficult concepts.

6. CAL offers a wide range of experiences that are otherwise not available to the student. It works as multimedia providing audio as well as visual inputs. It enables the student to understand concepts clearly with the use of stimulating techniques such as animation, blinking, graphical displays etc.

7. CAL provides a lot of drilling which can prove useful for low aptitude students and through which high-aptitude students can be escaped.

8. CAL can enhance reasoning and decision-making abilities.

b) Disadvantages

1. A CAL package may be regarded simply as a novelty, rather than an integral part of the educational process. It may threaten the objectives of the package.

2. Though simulation permits execution of chemical and biological experiments, hands-on experience is missing. Moreover, CAL packages cannot develop manual skills such as handling an apparatus, working with a machine etc.

3. There are real costs associated with the development of CAL systems. It is expensive in terms of staff time to devise and programme effective CAL.

4. Content covered by a certain CAL package may become outdated. A very high cost is involved in the development of these packages. If the course is outdated, the resources involved in its development will be a waste.

5. CAL packages may not fulfill expectations of teachers. Objectives and methods decided by the CAL author and of a teacher may differ.

6. Motivating and training teachers to make use of computers in education is a challenging task. They may have fear of this new device. They may be unwilling to spend extra time for preparation, selection and use of CAL packages. It may also be perceived as a threat to their job.

7. There are administrative problems associated with computer installation. The problems particularly related to the physical location of the computer resources, the cost of hardware maintenance and insurance and time-tabling.

8. The rapid development of hardware makes it difficult to select a system before it becomes obsolete. If a new system is installed by a maximum number of institutions, they may not get courseware required for the system and courseware developed so far may become useless.

Drill Questions:

1. Define the following terms of online Internet services giving TWO advantages and disadvantages of each: (a) E- Commerce, (b) E- Government, (c) E-Learning, (d) E-Banking (**Q9i, CGCE 2016**)

2. E-commerce is carried out online. State three possible effects on society if E-commerce increases greatly. (**Q1iii, CGCE 2014**)

3. Describe how transfer of funds is accomplished in E-commerce. (**Q8iv, CGCE 2015**)

CHAPTER 12: DATABASE DESIGN

12.1 Introduction

A database is an organized collection of related data stored in a way that it can be easily retrieved and manipulated. A telephone directory, a library catalogue and a class register are examples of manual or paper-based database systems. A paper-based database requires much paper as the database becomes larger, making it difficult to manipulate the database. The problems caused by paper-based systems are solved by the development of computer-based systems.

The capacity for computers to store large amounts of data and their ability to quickly and efficiently retrieve the data makes them ideal for creating and using electronic or computerized databases. A computerized database refers to a collection of related files that are digitized. Computerized databases are created using database software called database management systems (DBMS).

12.2 Database Models

12.2.1 Flat-file Database

A flat file database is a single table database, with separate copies of data in each part of the business. An example is a phone directory.

The problems encountered with flat file databases are

i) ***Data duplication***: data is repeated and hence stored many times. This wastes disk space and slows down query time.

ii) Maintenance is difficult as every occurrence of a piece of data needs to be updated if its value changes

iii) More manual data entry is required and therefore a greater likelihood of errors when data is being entered.

The solution to these problems is to divide the data into logical groups and store the data in multiple tables, then connect (relate) the tables to each other. This results to a ***Relational database***.

12.2.2 Relational Database Model

(RDBMS - relational database management system) A database based on the relational model developed by E.F. Codd. A relational database allows the definition of data structures, storage and retrieval operations and integrity constraints. In such a database the data and relations between them are organized in tables. A table is a collection of records and each record in a table contains the same fields.

12.2.3 Properties of Relational Tables

i) Values Are Atomic

ii) Each Row is Unique

iii) Column Values are of the Same Kind

iv) The sequence of columns is insignificant

v) The Sequence of Rows is Insignificant

vi) Each Column Has a Unique Name

Certain fields may be designated as keys, which means that searches for specific values of that field will use indexing to speed them up. Where fields in two different tables take values from the same set, a join operation can be performed to select related records in the two tables by matching values in those fields. Often, but not always, the fields will have the same name in both tables. For example, an "orders" table might contain (customer-ID, product-code) pairs and a "products" table might contain (product-code, price) pairs so to calculate a given customer's bill you would sum the prices of all products ordered by that customer by joining on the product-code fields of the two tables. This can be extended to joining multiple tables on multiple fields. Because these relationships are only specified at retreival time, relational databases are classed as dynamic database management system. The RELATIONAL database model is based on the Relational Algebra.

12.2.4 Object/Relational Model

Object/relational database management systems (ORDBMSs) add new object storage capabilities to the relational systems at the core of modern information systems. These new facilities integrate management of traditional fielded data, complex objects such as time-series and geospatial data and diverse binary media such as audio, video, images, and applets. By encapsulating methods with data structures, an ORDBMS server can execute comple x analytical and data manipulation operations to search and transform multimedia and other complex objects.

As an evolutionary technology, the object/relational (OR) approach has inherited the robust transaction- and performance-management features of its relational ancestor and the flexibility of its object-oriented cousin. Database designers can work with familiar tabular structures and data definition languages (DDLs) while assimilating new object-management possibilities. Query and procedural languages and call interfaces in ORDBMSs are familiar: SQL3, vendor procedural languages, and ODBC, JDBC, and proprietary call interfaces are all extensions of RDBMS languages and interfaces. And the leading vendors are, of course, quite well known: IBM, Inform ix, and Oracle.

12.2.5 Object-Oriented Database Model

Object DBMSs add database functionality to object programming languages. They bring much more than persistent storage of programming language objects. Object DBMSs extend the semantics of the C++, Smalltalk and Java object programming languages to provide full-featured database programming capability, while retaining native language compatibility. A major benefit of this approach is the unification of the application and database development into a seamless data model and language environment. As a result, applications require less code, use more natural data modelling, and code bases are easier to maintain. Object developers can write complete database applications with a modest amount of additional effort.

According to Rao (1994), "The object-oriented database (OODB) paradigm is the combination of object-oriented programming language (OOPL) systems and persistent systems. The power

of the OODB comes from the seamless treatment of both persistent data, as found in databases, and transient data, as found in executing programs."

In contrast to a relational DBMS where a complex data structure must be flattened out to fit into tables or joined together from those tables to form the in-memory structure, object DBMSs have no performance overhead to store or retrieve a web or hierarchy of interrelated objects. This one-to-one mapping of object programming language objects to database objects has two benefits over other storage approaches: it provides higher performance management of objects, and it enables better management of the complex interrelationships between objects. This makes object DBMSs better suited to support applications such as financial portfolio risk analysis systems, telecommunications service applications, World Wide Web document structures, design and manufacturing systems, and hospital patient record systems, which have complex relationships between data.

12.2.6 Hierarchical Database Model

The hierarchical data model organizes data in a tree structure. There is a hierarchy of parent and child data segments. This structure implies that a record can have repeating information, generally in the child data segments. Data in a series of records, which have a set of field values attached to it. It collects all the instances of a specific record together as a record type. These record types are the equivalent of tables in the relational model, and with the individual records being the equivalent of rows. To create links between these record types, the hierarchical model uses Parent Child Relationships. These are a 1:N mapping between record types. This is done by using trees, like set theory used in the relational model, "borrowed" from maths. For example, an organization might store information about an employee, such as name, employee number, department, salary. The organization might also store information about an employee's children, such as name and date of birth. The employee and children data forms a hierarchy, where the employee data represents the parent segment and the children data represents the child segment. If an employee has three children, then there would be three child segments associated with one employee segment. In a hierarchical database the parent-child relationship is one to many. This restricts a child segment to having only one parent segment. Hierarchical DBMSs were popular from the late 1960s, with the introduction of IBM's Information Management System (IMS) DBMS, through the 1970s

12.2.7 Associative Model

The associative model divides the real-world things about which data is to be recorded into two sorts:
Entities are things that have discrete, independent existence. An entity’s existence does not depend on any other thing. Associations are things whose existence depends on one or more other things, such that if any of those things ceases to exist, then the thing itself ceases to exist or becomes meaningless.
An associative database comprises two data structures:
i) A set of items, each of which has a unique identifier, a name and a type.
ii) A set of links, each of which has a unique identifier, together with the unique identifiers of three other things, which represent the source, verb and target of a fact that is recorded about

the source in the database. Each of the three things identified by the source, verb and target may be either a link or an item.

12.2.8 Entity Attribute Value (EAV) data model

The best way to understand the rationale of EAV design is to understand row modelling (of which EAV is a generalized form). Consider a supermarket database that must manage thousands of products and brands, many of which have a transitory existence. Here, it is intuitively obvious that product names should not be hard-coded as names of columns in tables. Instead, one stores product descriptions in a Products table: purchases/sales of individual items are recorded in other tables as separate rows with a product ID referencing this table. Conceptually an EAV design involves a single table with three columns, an entity (such as an olfactory receptor ID), an attribute (such as species, which is actually a pointer into the metadata table) and a value for the attribute (e.g., rat). In EAV design, one row stores a single fact. In a conventional table that has one column per attribute, by contrast, one row stores a set of facts. EAV design is appropriate when the number of parameters that potentially apply to an entity is vastly more than those that actually apply to an individual entity.

12.3 Database Modeling

12.3.1 Database Normalization

Database normalization is the process of organizing the fields and tables of a relational database to minimize redundancy and dependency. Normalization usually involves dividing large tables into smaller and less redundant tables and defining relationships between them. Normalization works through a series of stages known as normal forms. In order to achieve one level of normal form, each previous level must be met.

12.3.1.1 First Normal Form

For a database to be in first normal form (1NF), the following rules have to be met for each table in the database

- There are no columns with repeated or similar data
- Each data item cannot be broken down any further.
- Each row is unique i.e. it has a primary key
- Each field has an unique name

'Atomic' is the word used to describe a data item that cannot be broken down any further.

Examples of atomic data

- An identity number
- National Insurance number NY344599
- ISBN book reference e.g.1-931841-62-4
- Stock code PN10B
- A first name, 'John'
- A surname, 'Hunt'
- A telephone number
- A school name: 'Government Bilingual High School Bafoussam'

- A complete description 'A fountain pen is a writing instrument'.

The point of the two last examples indicates that 'atomic' does not mean 'single word'. It simply means that it would make no sense to reduce the data item any further as it would lose its meaning.

Examples of non-atomic data

- A full name: reginald longfor when a first name and surname is present in the database
- A full address: 6 Picton Road, London, WR1 4PG
- Data that are part of a larger dataset e.g Oxford Book club and Coventry Health club

In the last example, the database in question is storing club details and there are many different kind of clubs (Health, Book, Chess etc) being stored. So it would make sense to split this data into {Location, Club type}. So the *context* of the database needs to be understood when deciding if data is atomic or not.

Example of repeating data

ID	First name	Surname	Telephone1	Telephone2	Telephone3
2	Tom	Smith	22323	45634	3456345

In this case the database is trying to store contact telephone numbers for each person. The designer has created three fields to hold telephone numbers. This is what is meant by 'repeating data'. The telephone numbers are the same kind of data.

Examples of 1NF:

Reminder: Rules for first normal form

- There are no columns with repeated or similar data
- Each data item cannot be broken down any further.
- Each row is unique i.e. it has a primary key
- Each field has an unique name

a) *Which of these tables are NOT in first normal form?*

1.

Title	Firstname	Surname	Full name	Address	City	Postcode
Mr	Tom	Smith	Tom Smith	42 Mill Street	London	WE13GW

2.

ID	IP Address	username	last accessed	Activity	Result	active
1003	198.168.1.5	Smith	20081021:14.10	Save file	success	y

3.

ItemID	Product	Description	Size	Colour	Colour	Colour
234	Shoe	High Heel	6	red	blue	brown

4.

StudentID	Firstname	Surname	SchoolID*	ClassID*
354	Tom	Smith	6	5F

Comments:

Table 1. *This is not in 1NF. There is no primary key defined and so this record cannot be guaranteed to be unique. Also Full name is redundant - data is not atomic - as it is simply a combination of Firstname and Surname.*

Table 2. *This is in at least 1NF. It has a primary key identified by the underline. The data is atomic. Each field has a unique name. There are no repeat data.*

Table 3. *This is not in 1NF. It has a primary key, so it passes that test, data is atomic - tick in the box, but the colour the shoe can come in is being repeated - and furthermore the fields have the same name - so not in 1NF*

Table 4. *This is in 1NF as it meets all the rules for the first normal form.*

Questions to ask yourself to spot 1NF

- Does it have a primary key?
- Are each field name unique?
- Is the data atomic?
- Are there repeating / redundant fields?

b) Suppose a designer has been asked to compile a database for a fan club web site. Fans visit the web site to find like-minded friends.

The entities to be stored are

This indicates that each band has many fans. Each person is a fan of only one group.

BAND	FAN
The attributes of band are: • band id • band name • musictype	The attributes of a fan are: • fan id • firstname • surname • email addresse(s)

The database needs to be in first normal form.

First Attempt

This is the first time this person has designed a database and is not really sure of how to go about it. He designs the FAN table and loads it with the following records

FanID	Firstname	Surname	BandID*	email
1	Tom	Smith	23	tm@fan.org
2	Mary	Holden	56	mh@fan.org , mary@myhome.org

He has correctly set up a primary key. He also used a foreign key to refer to the band. But this is not in 1NF because Mary has two email addresses loaded into the email field. The data is not atomic. Loading data in this way is also going to make it very difficult to extract email addresses. Also the data length of the email field now has to be long enough to handle many email addresses, this is very inefficient and would be a problem if exceeded.

Second Attempt

He soon realises this is not a good idea. So then he decides to create two email fields

FanID	Firstname	Surname	BandID*	email	email2
1	Tom	Smith	23	tm@fan.org	

2	Mary	Holden	56	mh@fan.org	mary@myhome.org

This is also a poor approach - note that email2 is not being used in Tom's record and so is causing wasted storage, so not 1NF which seeks to avoid wasted / redundant data. Another problem is what if a fan has many more emails? Adding more and more email fields will make the wasted storage even worse.

Another problem is that the query to extract email addresses is far more complex than it needs to be as it has to examine each email field.

Solution

After trying out various ideas, he comes up with a good solution - create another entity called 'email' and use a foreign key in that table to link the fan and email tables together. The ER diagram is as follows:

The ER diagram shows that each fan can have many emails, but an email can only belong to one fan.

The FAN and EMAIL table now look like this

FAN

FanID	Firstname	Surname	BandID*
1	Tom	Smith	23
2	Mary	Holden	56

EMAIL

EID	FanID*	email
1	1	tm@fan.org

2	2	mh@fan.org
3	2	mary@myhome.org

Mary (FanID = 2) has two entries in the email table. There is no problem adding even more emails for her. Extracting emails is now simple as there is only one email column. There is no wasted storage.

The tables are now in first normal form (1NF) as they obey the following rules

- *Each table has a primary key*
- *Each field name is unique*
- *Data is atomic*
- *No repeating / redundant fields*

12.3.1.2 Second Normal Form

Most tables tend to have a single-attribute (i.e. simple) primary key. Like this

CUSTOMER

ID	Firstname	Surname	Telephone	email
2	Tom	Smith	22323	ts@aol.com

But sometimes a table has a primary key made up of more than one attribute i.e. it has a *compound primary key*.

CONCERT

Venue	Artist	Attendance	Profit	Style
Wembley	Girls Aloud	53000	12334	Girl band
NEC	Leona Lewis	45000	66433	Female soloist

The table above is using both the venue and artist as the compound primary key.

It is in this situation that the extra rule for second normal form comes in handy. The rule states

- ***Non-key attributes must depend on every part of the primary key***
- *The table must already be in first normal form*

So inherently, any table that is already in 1NF and has a simple primary key is automatically in second normal form as well.

Consider the Concert example above - this is NOT in second normal form. Notice the attribute called Style. This is describing the style of artist - it has nothing to do with where the concert took place! And so its value does not depend on EVERY part of the primary key, so the rule for second normal form is not being met.

The reason for this rule is to ensure there is no redundant data being stored.

For example, let's add another Girls Aloud concert to the table

Venue	Artist	Attendance	Profit	Style
Wembley	Girls Aloud	53000	12334	Girl band
NEC	Leona Lewis	45000	66433	Female soloist
NEC	Girls Aloud	76090	53789	Girl band

Notice that the 'girl band' value is being repeated and so is causing the database to be bigger than it needs to be.

Going on with the concert table

CONCERT

Venue	Artist	Attendance	Profit	Style
Wembley	Girls Aloud	53000	12334	Girl band
NEC	Leona Lewis	45000	66433	Female soloist
NEC	Girls Aloud	76090	53789	Girl band

This table needs to be split so that non-dependent attributes are removed and only stored once.

In this case a 'style' table is formed that has Artist as the simple primary key

CONCERT

Venue	Artist	Attendance	Profit
Wembley	Girls Aloud	53000	12334
NEC	Leona Lewis	45000	66433
NEC	Girls Aloud	76090	53789

STYLE

Artist	Style
Girls Aloud	Girl band

Leona Lewis	Female soloist

Now the rule for 2NF is being met by both tables - every non-key attribute is depending on the complete primary key. There is no redundant data.

Of course there could be more than one attribute related to different parts of the primary key. Consider a table like this:

CONCERT

Venue	Artist	Date	Attendance	Profit	City	No1Hits	Style
Wembley	Girls Aloud	1/10/09	53000	12334	London	5	Girl band
NEC	Leona Lewis	1/10/09	45000	66433	Birmingham	2	Female soloist
NEC	Girls Aloud	7/11/09	76090	53789	Birmingham	5	Girl band

As before the Style attribute only depends on Artist, but now No1Hits also only depends on the Artist. This table also includes City and this only depends on the Venue.

So to make this database into second normal form, four tables need to be created

CONCERT

VenueID	Artist	Date	Attendance	Profit
005	0112	1/10/09	53000	12334
006	0115	1/10/09	45000	66433
006	0112	7/11/09	76090	53789

STYLE

Style ID	Style
001	Girl band
002	Solo artist
003	Rap

ARTIST

ArtistID	Artist	No1Hits	StyleID
0112	Girls Aloud	20	001

0115	Leona Lewis	3	002

VENUE

Venue ID	Venue	City
005	Wembly	London
006	NEC	Birmingham

Summary

The rules for second normal form are

- ***Non-key attributes must depend on every part of the primary key***
- *The table must already be in first normal form*

12.3.1.3 Third Normal Form

For a database to be in third normal form, the following rules have to be met

- *It is already in 2NF*
- ***There are no non-key attributes that depend on another non-key attribute***

What this is trying to do is to spot yet another source of redundant data. If the value of an attribute can be obtained by simply making use of another attribute in the table, then it does not need to be there. Loading that attribute into another table and linking to it will make the database smaller.

To clarify, consider the table below

CONCERT

Venue	Artist	Date	Attendance	Profit	City	Country
Wembley	Girls Aloud	1/10/08	53000	12334	London	UK
NEC	Leona Lewis	1/10/08	45000	66433	Birmingham	UK
Carnegie Hall	Girls Aloud	7/11/08	76090	53789	New York	USA

Notice that the country could be obtained by referring to the City - if the concert was in London then you know it is also in the UK - no need to look at the primary key!

So to make this database into third normal form, split the table as follows

CONCERT

Venue	Artist	Date	Attendance	Profit	City*

Wembley	Girls Aloud	1/10/08	53000	12334	London
NEC	Leona Lewis	1/10/08	45000	66433	Birmingham
Carnegie Hall	Girls Aloud	7/11/08	76090	53789	New York

COUNTRIES

City	Country
London	UK
Bimingham	UK
New York	USA

The new table called COUNTRIES has City as the primary key and country as an attribute. The Concert table has City as a foreign key. So now you can obtain the country in which any particular concert took place and there is no redundant data.

Reminder, 3NF means:

- *It is already in 2NF*
- ***There are no non-key attributes that depend on another non-key attribute***

Example 1

CUSTOMER

CustomerID	Firstname	Surname	City	PostCode
12123	Harry	Enfield	London	SW7 2AP
12443	Leona	Lewis	London	WC2H 7JY
354	Sarah	Brightman	Coventry	CV4 7AL

This is not in strict 3NF as the City could be obtained from the Post code attribute. If you created a table containing postcodes then city could be derived.

CustomerID	Firstname	Surname	PostCode*
12123	Harry	Enfield	SW7 2AP
12443	Leona	Lewis	WC2H 7JY
354	Sarah	Brightman	CV4 7AL

POSTCODES

PostCode	City
SW7 2AP	London
WC2H 7JY	London
CV4 7AL	Coventry

Example 2.

VideoID	Title	Certificate	Description
12123	Saw IV	18	Eighteen and over
12443	Igor	PG	Parental Guidance
354	Bambi	U	Universal Classification

The Description of what the certificate means could be obtained frome the certifcate attribute - it does not need to refer to the primary key VideoID. So split it out and use the primary key / secondary key approach.

Example 3

CLIENT

ClientID	CinemaID*	CinemaAddress
12123	LON23	1 Leicester Square. London
12443	COV2	34 Bramby St, Coventry
354	MAN4	56 Croydon Rd, Manchester

CINEMAS

CinemaID	CinemaAddress
LON23	1 Leicester Square. London
COV2	34 Bramby St, Coventry

MAN4	56 Croydon Rd, Manchester

In this case the database is almost in 3NF - for some reason the Cinema Address is being repeated in the Client table, even though it can be obtained from the Cinemas table. So simply remove the column from the client table

Example 4

ORDER

OrderID	Quantity	Price	Cost
12123	2	10.00	20.00
12443	3	20.00	60.00
354	4	30.00	120.00

In this case the cost of any order can be obtained by multiplying quantity by price. This is a 'calculated field'. The database is larger than it needs to be as a query could work out the cost of any order. So to be in strict 3NF you would remove the Cost column.

12.3.1.4 Benefits of Normalisation

1. The database does not have redundant data, it is smaller in size so less money needs to be spent on storage

2. Because there is less data to search through, it is much faster to run a query on the data

3. Because there is no data duplication there is better data integrity and less risk of mistakes.

4. Because there is no data duplication there is less chance of storing two or more different copies of the data

5. Once change can be made which can instantly be cascaded across any related records

12.3.1.5 Problems of Normalisation

1. You need to be careful with trying to make data atomic. Just because you can split some types of data further, it isn't always necessarily the correct thing to do. For example, telephone number might contain the code followed by the number 01234 567890. It wouldn't be sensible to separate out these two items.

2. You can end up with more tables than an unnormalised database

3. The more tables and the more complex the database, the slower queries can be to run

4. It is necessary to assign more relationships to interact with larger numbers of tables

5. With more tables, setting up queries can become more complex

12.3.2 Entity-Relationship Modelling

When a relational database is to be designed, an entity-relationship model is drawn at an early stage and developed as the requirements of the database and its processing become better understood.

***Definition**: An E-R model is a diagram which uses basic graphic symbols to show the organization of and relationships between data in a database.*

An E-R diagram serves as a schema diagram for the required database. A schema diagram is any diagram that attempts to show the structure of the data in a database.

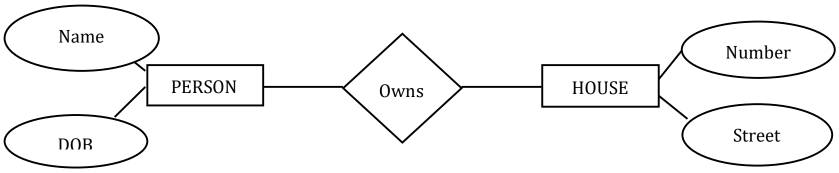


Figure 12.1: *An E-R diagram*

The basic elements of an E-R diagram are entity sets, attributes and relationship types.

12.3.2.1 Entity Set

An entity is a person, place, concept or thing for which we intend to collect data. For example, a customer, an employee, a book, an appointment.
A group of entities that share the same properties is an entity set. An entity is therefore a member or an instance of an entity set. In an E-R diagram, an entity set is represented by a rectangle. In the above E-R diagram, PERSON and HOUSE are entity sets

12.3.2.2 Attribute

An attribute is a fact about an entity or a property that describes an entity. For example, a person's name, date of birth or gender, a vehicle's model, color or brand. Attributes store the actual data we want to keep about each entity within an entity set. An attribute is represented by an ellipse. In the above E-R diagram, name and date of birth are attributes of the entity set PERSON while Number and Street are attributes of the entity set HOUSE.

12.3.2.3 Relationship Type

A relationship type is a named association between entities. A person (entity) owns (relationship) a house (entity), a teacher (entity) teaches (relationship) a subject (entity). Normally, individual entities have individual relationships of the type between them but in an E-R diagram, this is generalized to entity sets and relationship types. For example, the entity

set PERSON is related to the entity set HOUSE by the relationship type OWNS.A relationship type is represented by a diamond.

A relationship type can be characterized by a degree and cardinality ratio.

a. Degree of a Relationship Type

The degree of a relationship type is the number of entity sets involved in the relationship. It can be unary, binary, ternary or n-ary.

i. A **unary relationship** type is one that involves entities from a single entity set. E.g. the relationship MANAGES between entities within the entity set EMPLOYEE.

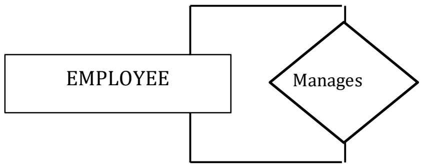


Figure 12.2: *Unary relationship diagram*

ii. A **binary relationship** type is a relationship between entities from two different entity sets. An example is the relationship OWNS in the E-R diagram above.

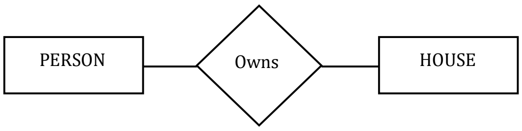


Figure 12.3: *Binary relationship diagram*

iii. A **ternary relationship** type is one that involves entities from three different entity sets. An example is a LECTURER who teaches a certain COURSE in a DEPARTMENT.

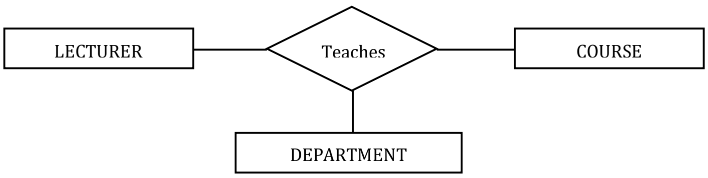


Figure 12.4: *Ternary relationship diagram*

iv. An $\boldsymbol{n}$**-ary relationship** type involves entities from n different entity sets.

b. Cardinality Ratio of a Relationship Type

Cardinality is the maximum number of entities within each entity type that can take part in a relationship.

For example, what is the maximum number of houses that a person can own? On the other hand, the minimum number of entities within each entity that can take part in the relationship is the Optionality. For example, what is the minimum number of houses that can be owned by a person?

The cardinality ratio is a ratio of the cardinalities of the entity sets involved in a relationship. It can be one-to-one (1:1), one-to-many (1:N) or many-to-many (M:N).

i) In the relationship OWNS between PERSON and HOUSE, a person can own zero or many houses. Therefore, the cardinality of PERSON in the relationship OWNS is many while the optionality is zero.
On the other hand, a house is owned by one person. The cardinality of HOUSE in the inverse relationship IS_OWNED_BY, is one while the optionality is one. The relationship OWNS is therefore described as a one-to-many (1:N) relationship.

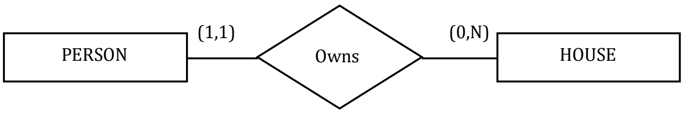


Figure 12.5: *(1:N) relationship*

ii) In the relationship RECEIVES between STUDENT and SLIP, each student receives one and only one result slip. The cardinality of student is one and the optionality is one. Each result slip is issued to one and only one student. The cardinality of SLIP is one and the optionality is one. This relationship is described as one-to-one (1:1).

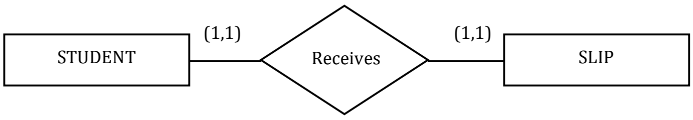


Figure 12.6:*(1:1)*

iii) In the relationship TEACHES between TEACHER and SUBJECT, a teacher teaches one or many subjects. The cardinality of TEACHER is many and the optionality is one.
A subject is taught by one or many teachers. The cardinality of SUBJECT in the inverse relationship IS_TAUGHT_BY, is many and the optionality is one. This relationship is described as many-to-many (M:N).

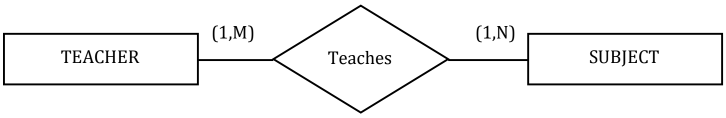


Figure 12.7:*(M:N) relationship*

Remark In a normalized database, many-to-many relationships are eliminated by creating a link entity between the entities involved in the relationship.
The above TEACHES relationship when normalized becomes

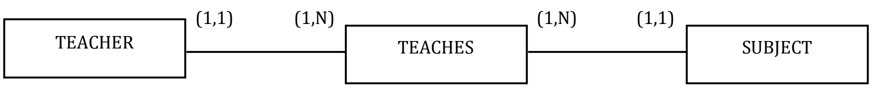


Figure 12.8: *Normalized (M,N) relationship*

The primary key of the link entity is a composite key that consists of the primary keys of the entities TEACHER and SUBJECT.

The above representation of an E-R diagram is the Chen Convention. Another way of representing E-R diagrams is the crow's foot notation that uses three symbols to show cardinality ratios. Here, a circle means *zero*, a line means *one* and a crow's foot means *many*. The cardinality is shown next to the entity type and the *optionality* (if shown at all) is shown behind it.

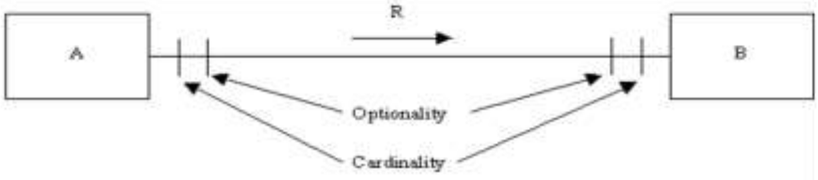


Figure 12.9: *Representation of E-R diagrams using crow's notation*

Where A and B are entity sets and R is the relationship type.

12.4 Database Management Systems

A database management system (DBMS) is a computer program that is used to create, manipulate and access a database. Examples are MS Access, MySQL, Oracle, Sybase, dBase, Paradox, Objecstore and O_2.

A DBMS provides users with:

a) ***Data Definition Language (DDL):*** is a syntax similar to a computer programming language for defining data structures, especially database schemas.
b) ***Data Manipulation Language (DML):*** is a family of syntax elements similar to a computer programming language used for selecting, inserting, deleting and updating data in a database. Performing read-only queries of data is sometimes also considered a component of DML. A language which changes the contents (instance) of the database
c) ***Data Control Language (DCL):*** is a syntax similar to a computer programming language used to control access to data stored in a database. Examples of DCL commands include:
- GRANT to allow specified users to perform specified tasks.
- REVOKE to cancel previously granted or denied permissions.

12.4.1 Microsoft Access 2007

MS Access 2007 is made up of several components including:

- ✓ Tables
- ✓ Forms
- ✓ Queries
- ✓ Reports

These components are called *database objects*. One or more of these components are formed when a database is created. These components are stored in a single database file.

12.4.1.1 Tables

A table is an area of the database where information is stored. Tables organize data into columns (*called fields*) and rows (*called records*). Example: a database storing information about books and authors for a library could have a "BOOKS" table to store information about books (title,

publisher, Year, etc) and an “AUTHORS” table to store information about authors (name, birth date, birth place, etc).

i. A field is a specific piece of data stored in a table. The above mentioned ‘BOOKS’ table has fields ‘Title’, ‘Publisher’, ‘Year’ etc.

ii. A record is an individual entry in a database table. It comprises of one or more fields, depending on the number of fields defined to the table. For example: if five books are entered in our books and authors database, the “BOOKS” table would have five records (one for each book).

iii. Each field in a database table is assigned a data type when the field is created. The data type determines the kind and format of the information stored in the field. Example: the “Title” field in the “BOOKS” table will store text that will vary with each book in the database, so the data type in this case will be text or character type.

a) Creating a New Table

To create a new table,

- *Click the Create Tab*
- *Select Table in the Tables group to create a new table in Datasheet view or Table Design to create a new table in Design view*

b) Adding Fields in a Table

To add a field to a table in Datasheet View,

- *Click the Add New Field column label.*
- *Click Rename in the Fields & Columns group.*
- *Type the field name and press Enter. Access creates the field. Continue until you have created all of the fields in your table.*
- *Press Enter without entering a field name to end your entries.*

To add a field to a table in Design View,

- *Select a cell in the Field Name column*
- *Type a name for the first field in the table and press Enter*
- *Select a data type and press Enter*
- *Type a description and press Enter*

c) Naming and Saving a Table

To name and save a table,

- Click the Save button on the Quick Access toolbar. The Save As dialog box appears.
- Type the name you want to give your table.
- Click OK. Access names your table.

Tip: *You can use the Rename option at any time to rename any column. For example, you can rename the ID column Author_ID.*

d) Setting a Primary Key

A primary key is a field in a table that allows each entry in the table to be identified in a unique way. No two entries in the database can have the same primary key. For example your ID card number is unique. By default, Access sets the first field in the table as the Primary Key field. To set a Primary Key:

- *Switch to Design View*
- *Position your cursor in the field you wish to set as the Primary Key*
- *Click the Primary Key button on the Ribbon*

e) The Foreign Key

The primary key is used to link the table with other tables in the database. When a primary key in one table (known as the source table) is linked to another table (known as the target table), the connecting field in the target table is called the foreign key. A foreign key must have the same structure, data type, and field size as the associated primary key, but it must not have the same name. Also, the foreign key in a relationship between two tables need not be a primary key in its own table.

12.4.1.2 Relationships

Relating tables reduces the unnecessary duplication of data (redundancy).The general procedure for creating relationships in Access is to

- *Determine the type of the relationship and identify the source and target tables*
- *Create the foreign key field(s) in the target table if they are not already present*
- *Select the Database Tools tab*
- *Click on the Relationships command*
- *Add all the tables involved in the relationship to the window*
- *Create the relationships by dragging the primary keys from the source table(s) and dropping them on the associated foreign key(s) in the target table(s).*

For example: Let's create a relation between the AUTHORS table and the BOOKS table.

12.4.1.3 Queries

A query is a command that tells the database what data the user wants to see. A query allows information to be found and retrieved from one or more tables based on a set of search conditions you define. For example: a query can be written specifying that the database should retrieve all the books from the "BOOKS" table written by William Shakespeare.

To create a query,

- *Click the Create tab on the Ribbon*
- *Click the Query Design command*
- *Double-click Create Query in Design View*
- *Select the table that you would like to base your Query on*
- *Click Add*
- *Close the Show Table window*

In order to control which records are displayed, you must define criteria in a Query.
To define criteria for your query:

- *Position your cursor in the criteria row in the field for which you wish to define the criteria for*
- *Enter the criteria*
- *Click the Run Query button*

For example: *a query to find the books written by Hilary Clinton*

The result of a query is called a *recordset*.

To Save the Query**:**

- *Click the Save Icon*
- *Enter a name for the query*
- *Click OK*

12.4.1.4 Forms

A form is an interface that allows users to enter and view data. Forms give the ability to choose the format and arrangement of fields. They can be used to enter, edit and display data. Forms can retrieve data from one or more tables, and display the output on the screen.

To create a Form using the Form Wizard

- *Navigate to the table you want to base the form on*
- *Click Create on the Ribbon*
- *Click Forms*

You are able to navigate using the navigation arrows at the bottom of the form.

To enter a record on the Form:

- *Click the View button on the Ribbon to switch from Layout View to Form View*
- *Enter the data for each field in the record, pressing the Enter key to move to the next field*
- *Press Enter after you have entered data for the last field*

NB. *The form feeds the table. If you edit a record on the form, or create a new record, that data will be passed to the table it is associated with.*

12.4.1.5 Reports

A report is an effective way to analyse and present data in printable format using a specific layout. Reports are the primary method of retrieving and viewing information. Using queries, reports pull information the user has requested and either print that information or display it on screen. The information that appears and the formatting and appearance of a report are set when the report is created.

To create a Report using the Report Wizard:

- *Click the Create tab on the Ribbon*
- *Click the Report Wizard command*
- *Select the table or query upon which the report will be based*
- *Select the fields that you want to include on the report by double clicking on them*
- *Click Next*
- *Select a style for the report*
- *Click Next*
- *Type a title for the report*

Click Finish

12.4.2 Structured Query Language

Databases contain data in the form of records within tables and a number of relationships between them. But there has to be a method of setting up a database in the first place and also a method of accessing the data according to some criteria.This is why SQL was developed.

SQL stands for *Structured Query Language*. SQL is now maintained as a standard language by the American National Standards Institute (or ANSI).A relational database management system (RDBMS) supports SQL as a control language.

Some RDBMS may have some custom SQL extensions peculiar to itself but it will support all the standard functionality that ANSI compliant SQL offers.There are many commercial RDBMS such as Oracle and DB2 and there are free open source RDBMS available such as MySQL.

12.4.2.1 Creating a Database and Table

The first command to issue within the RDBMS is to set up a database. This is quite straightforward, namely

CREATE DATABASE test

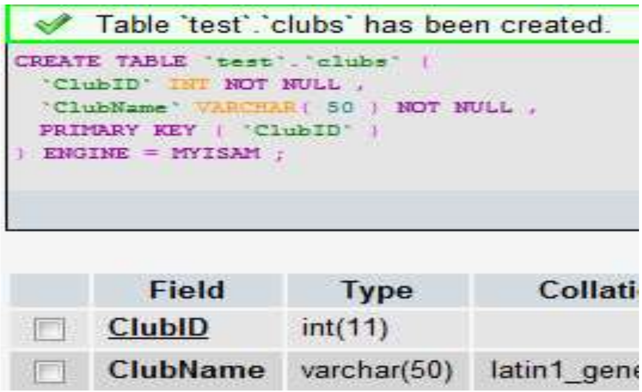
This SQL command will cause the RDBMS to create the supporting structures for a database called 'test'.

At this point it is an empty database.

The next step is to create a table within the database. And this is done as follows

```
CREATE TABLE Clubs ( ClubID INT NOT NULL, ClubName VARCHAR(50) NOT
NULL, PRIMARY KEY (ClubID))
```

The figure below shows a real table being created in open source *MySQL* with the free admin tool called *PHPMyAdmin*



The top part shows the SQL generated to create the table and the lower part shows the resulting table structure.

Let's take each element at a time.

The first command is the standard SQL command to create a table CREATE TABLE

This is followed by the table's name 'Clubs' and then a set of parameters within a pair of brackets

```
CREATE TABLE Clubs (.....)
```

The parameters within the brackets define, as a minimum, each field name and their data type. For example the first parameter in this example is 'ClubID' and it is of type Integer.

```
CREATE TABLE Clubs (ClubID INT .....
```

The second field is 'ClubName' and is of type VarChar. The length of VarChar is 50 characters.

```
CREATE TABLE Clubs (ClubID INT, ClubName VARCHAR(50) ....
```

A comma separates each field declaration.

SQL also support some additional declarations. Note the PRIMARY KEY in the example. This tells the system that the primary key is the field ClubID. In the figure above the ClubID is underlined to show that it is the primary key of the table

You can insist on the field value not being empty by using the term NOT NULL. So the first field ends up as

```
CREATE TABLE Clubs (ClubID Integer NOT NULL,
```

Regarding the VarChar, notice that 50 elements of storage has been allocated to this field. When designing a field you should think carefully about the expected length of the field values - too short and the values may become truncated, too long and you could waste a lot of space.

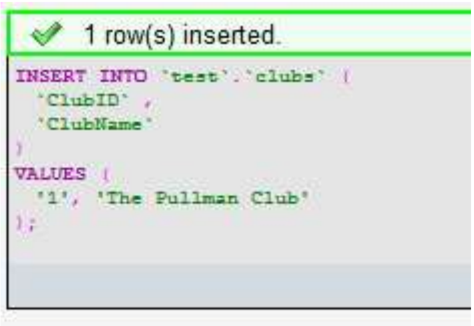
12.4.2.2 Inserting values into a Table

Once an empty table has been created, the next thing is to load some records into it. This is done using the sql INSERT command

Carrying on with the example of the clubs table, the command below will insert one record into the table

INSERT INTO \`test\`.\`clubs\` (\`ClubID\`, \`ClubName\`) VALUES ('1', 'The Pullman Club');

The figure below shows a nicely formatted view of inserting a row into MySQL



Breaking it down, the first command is

```
INSERT INTO test.clubs .....
```

This identifies the database and the table to be used with the insert command. (This is a 'boots and braces' approach - if there is no other database open then you can just use the table name rather than the full database definition)

Next comes

```
INSERT INTO `test`.`clubs` (`ClubID`, `ClubName`)....
```

The items within the brackets are the fields that will be loaded with new data

Next the sql term VALUES appears making way for the actual values

```
INSERT INTO `test`.`clubs` (`ClubID`, `ClubName`) VALUES ...
```

Along come the values themselves ...

```
INSERT INTO `test`.`clubs` (`ClubID`, `ClubName`) VALUES ('1',
'The Pullman Club');
```

NB:*In practice you have to be very careful that all the data types are correct and in the right order with nothing missing, most RDBMS are very unforgiving if you do not have it completely correct!*

12.4.2.3 Updating a Table row

Now that the table exists and has at least one row (record), you can use the sql UPDATE command to change a record.

Carrying on with the example from the previous page, we now want to alter the existing record containing 'The Pullman Club' and to change it to 'The Truman Club'

The existing row is shown below where the record is identified by the primary key ClubID = 1

ClubID	ClubName
1	The Pullman Club

To do this the following sql command is issued

```
UPDATE `test`.`clubs` SET `ClubName` = 'The Truman Club' WHERE
`clubs`.`ClubID` =1;
```

The nicely formatted MySQL command is shown below

1 row(s) affected.

```
UPDATE `test`.`clubs` SET `ClubName` = 'The Truman Club' WHERE `clubs`.`ClubID` =1;
```

Breaking this down step by step

The first part is the sql command UPDATE followed by the fully defined table 'clubs'

```
UPDATE `test`.`clubs` ....
```

Then comes the word SET to identify the fields to be updated

```
UPDATE `test`.`clubs` SET `ClubName` = 'The Truman Club' ...
```

then comes the extremely important WHERE clause which pins down exactly which records are to be updated

```
UPDATE `test`.`clubs` SET `ClubName` = 'The Truman Club' WHERE
`clubs`.`ClubID` =1;
```

In this case the WHERE clause is declaring that only the record whose primary key is ClubID =1 is to be updated. This means only one record will be changed.

In summary, the generic format of the UPDATE command is

```
UPDATE {table} SET {field value = new value} WHERE {selection
criteria}
```

Practical advice: If you get the WHERE bit wrong in an update query, then you could easily overwrite every record! On a production server with a live database, it is advisable to use the SELECT query first to check out what the WHERE criteria returns - Select is safe because it does not alter any records. Once you are confident with the selection criteria then you can use the update command.

12.4.2.4 Selecting Records and reporting

Once the database has been set up and the tables populated with records, it is time to use the database in earnest by selecting records out of it. The returned results can be used to generate nicely formatted reports or they can be used to process the data in some way.

The most widely used command to extract records from a database is the SELECT command. The example we have used has been expanded slightly to include three records

ClubID	ClubName
1	The Truman Club
2	The Benedict Club
3	The Norbury chess club

12.4.2.4.1 All records

If we wanted to extract ALL records from the table, the following command is issued

```
SELECT * FROM `clubs`
```

The command begins with the word SELECT, then a star * symbol which means 'all fields' in sql followed by the FROM statement and a table name. The generic command is

```
SELECT {fields to be extracted} FROM {table}
```

The command extracts all records because there is no WHERE clause to qualify the records. This is quite a useful command of itself if you intend to process every record in the defined table in some way.

12.4.2.4.2 Extracting a single record

The sql SELECT command can be extended in order to extract records that meet a specific criteria.

Example: extracting a single record

The easiest way to do this is to target the primary key with a single value. Like this

```
SELECT * FROM `clubs` WHERE `ClubID`=2
```

This returns a single record as shown below

ClubID	ClubName
2	The Benedict Club

The new part of the command is the WHERE clause which in this case only selects the record whose primary key is 2

12.4.2.4.3 Extracting a number of records

We have described how to pull all records (absent WHERE clause) and we have described how to extract a single record (WHERE includes single value primary key).

But another very common requirement is to extract a sub-set of the table records.

For instance the command

```
SELECT * FROM `clubs` WHERE `ClubID`>1
```

This has a WHERE clause that will select any record whose primary key is greater than one. The result in the example is:

ClubID	ClubName
2	The Benedict Club
3	The Norbury chess club

The WHERE clause of ansql query can be very complicated in order to extract some specific records.

In the example so far, we have three records

ClubID	ClubName
1	The Truman Club
2	The Benedict Club
3	The Norbury chess club

One of the basic requirements of a database is to be able to delete a record

The sql command to do this is

```
DELETE FROM clubs WHERE ClubID = 1
```

The DELETE FROM is qualified with a table name

```
DELETE FROM clubs
```

followed by a WHERE clause to limit the number of records to be deleted

```
DELETE FROM clubs WHERE ClubID = 1
```

In this case a single record is to be deleted from the clubs table whose primary key is 1. The WHERE clause can be more general than this in order to select a specific set of records.

12.4.3 Further SQL

The syllabus is mainly concerned with the fundamental sql queries of CREATE, INSERT, UPDATE, SELECT and DELETE. And the examples demonstrate their use with a *single* table. But a full relational database is usually comprised of several related tables. SQL supports the extraction of records based on the values across more than one table.

A key command is the JOIN. This allows records selection, insertion and updates to occur based on criteria across a number of tables. If you wish to find out more about sql then we recommend you explore the uses of this command as well

Drill Questions

1. A database will be made to store information about patients in a hospital. On arrival, each patient's personal details (name, address, and telephone number) are recorded where possible, and they are given an admission number. They are then assigned to a particular ward (Accident and Emergency, Cardiology, Oncology, etc.). In each ward there are a number of doctors and nurses. A patient will be treated by one doctor and several nurses over the course of their stay, and each doctor and nurse may be involved with several patients at any given time.
From the description, draw a corresponding E-R diagram showing all entity sets, attributes, relationship types and cardinality ratios. (**CGCEB**)

2.(i) Explain the following terms. (**West Regional Mock CGCE 2016**)
a) DBMS
b) Relational database
c) Flat file database
(ii) What are the benefit (three) database normalisation?
(iii) Put the table below in its first and second normal forms.

Contacts						
Name	**Company**	**Address**	**Phone1**	**Phone2**	**Phone3**	**ZipCode**
Joe	ABC	123	5532	2234	3211	12345

Jane	XYZ	456	3421			14454
Chris	PDQ	789	2341	6655		14423

3. Given a library, you are asked to design a database such that in this library:
- A customermay borrow a book or more.
- For every book there is just one author.
- An author could have written many books.
- When borrowing abook, the code of the book, the borrower's code, the borrow date, and the return date must be recorded as well as the state of the book.
- When the borrower returns the book, the same information as on the bottow date is recorded.

Answer the following questions given that the entities Author, Book and Customer have attributes as shown below:

Author(Aut-code, Name)
Book (B-code, title, edition)
Customer (Reg-No, Name, Address)

(i) Give an entity relationship model for the library. Clearly show the relationships between the entities.

(ii)(a) What are the primary keys of Author, Book and Customer?

(b) Write, using the same notation as that for the entities given above, the following relations of obvious meanings: BorrowBook, WriteBook and ReturnBook. Underline their primary keys.

(c) Select any three relations or entities and say whether or not they are in 3rd Normal form (3NF). (**CSC Q3, CGCE 2013**)

CHAPTER 13: ALGORITHMIC DESIGN AND PROGRAMMING

13.1 Data Types and Structures

13.1.1 Data Types

Computers manipulate data of different types like numbers, letters, sound and images. No matter the type, each data is stored as a pattern of binary digits. Numbers can be added, subtracted, divided and multiplied whereas letters cannot. This implies that data of a given type will be treated differently from data of another type. As such, the type of any data stored in the computer must be specified so that the correct operations will be performed on it. A data type is defined as a set of values and a collection of operations on those values. Examples of data types are integer, real, string and Boolean.

a. Integer

An integer is a whole number that can be positive, negative or zero. Examples: -3, 5, 67, -134 and 4231.

b. Real

A real is a number that contains a decimal point. Computers manipulate numbers with fractions as floating point numbers. Examples: 3.14, 12.25 and 0.001.

c. Boolean

A Boolean data type stores one of only two values – true or false, yes or no, or on or off. Usually these values are stored as 1 for true and 0 for false.

d. Character

A character is any letter, digit, space, punctuation mark, or symbol that can be typed on a computer. The word "computer" for example, consists of eight characters. The phrase "data type" takes up nine characters. A character data type stores only a single character or digit. Each character requires one byte of space, so "computer" takes up 8 bytes ($8 \times 8 = 64\,bits$). Examples: A, 5, -, and !.

e. String/Text

A string is a set of characters grouped together. A string is sometimes just referred to as ‘text’. It can contain a mixture of numbers, letters and, spaces. Numeric data stored as text is only for representation not for calculation. Any type of alphabetic or numeric data can be stored as a string. Examples: “Limbe City”, “10base-2” and “36.85”.

f. Date/ Timestamp

Date and timestamp are data types for storing date and time respectively. The format used for date is either dd-mm-yy or mm-dd-yy and hh:mm:ss or ss:mm:hh for time.

g. Container

A container is a data type used for storing images, video, sound or another type of 'complex' file. In some languages, binary large object (blob) is used rather than container.

13.1.2 Data Structures

A data structure is a way of organizing and storing data in memory so that it can be used efficiently. Any data structure is designed to organize data so that it can be accessed and worked with in appropriate ways. A well-designed data structure allows a variety of critical operations to be performed, using as few resources, both execution time and memory space, as possible. Data structures include arrays, records, stacks, queues, trees, graphs and hash tables.

13.1.2.1 Arrays

An array is a data structure, which allows a set of items of identical data type to be stored together using the same identifier name. When an array is created, its dimension (size) and the type of its elements are specified.

For example: **String** Names[5]
Names :**Array** [1..5] of **string**

The above statements create an array called "Names" of five elements in C and Pascal respectively. The size of the array is 5 and its elements are of type string.
To reference an element in an array, an index is used. An index is an integer that gives the element at the indexed position in the array.
For example in C, Names[0] references the first element in the array Names, Names[1] references the second element, Names[2] references the third and so on…
In Pascal, the referencing starts from 1 rather than 0.

a. One-dimensional Arrays

A one-dimensional array is an array which is declared using a single index and can be visually represented as a list. The array "Names" above is one-dimensional.
If the array "Names" contains five elements (names), Raissa, Peter, James, Leticia and Isaac, it can be visually represented as follows:

Element	Raissa	Peter	James	Leticia	Isaac
Index	[0]	[1]	[2]	[3]	[4]

Names[0] == Raissa, Names[4] == Isaac

b. Two-dimensional Arrays

A two-dimensional array is an array which is declared using two indices and can be visually represented as a table.
For example:

int Tab[3][2] or Tab : **Array** [1..3, 1..2] of **integer**
3 is the number of rows and 2 is the number of columns in the table Tab.

The elements of Tab will be referenced as:
Tab[0][0] Tab[0][1]
Tab[1][0] Tab[1][1]
Tab[2][0] Tab[2][1]
Example: Given the 2-dimensional array Tab[4][2] below, what are the elements given by the references Tab[0][0], Tab[1][1], Tab[3][0] and Tab[2][1]?

Index	[0]	[1]
[0]	5	3
[1]	1	4
[2]	3	1
[3]	2	8

Each individual element can be referenced by its row and column indices. For example:
Tab[0][0] = 5
Tab[1][1] = 4
Tab[3][0] = 2

13.1.2.2 Record

A record is a data structure which allows items of different types that define a particular structure or object, to be stored together. The elements of a record are usually called *fields* or *members*.
Defining a record includes specifying the data type of each field and a name (label) by which the field can be accessed.
For example,

```
Date  = RECORD
  Day: Integer
  Month: String
  Year: Integer
End RECORD
```

```
Student = RECORD
  Name: String
  Gender: Character
  DOB: Date
End RECORD
```

```
struct  Student
  {
string   Name
char     Gender
     Date     DOB
  }
```

The above Date record has a numeric field Day, a text field Month and a numeric field Year. The Student record has fields Name, Gender and DOB which is of type Date. The table below gives examples of student records.

Name	Gender	DOB
Raissa	F	01 Apr 1996
Peter	M	17 Oct 1995
James	M	05 May 1995
Leticia	F	09 Dec 1997
Isaac	M	29 Feb 1995

Records

Each row in the table corresponds to a different record. This table has 5 records.
A field in a record is referenced as follows:

RecordName.fieldName = “”
Raissa.DOB = 01 Apr 1996

Record data structures are always used in association with arrays. That is, we define an array whose elements are of type record. When this is done, we have an *array of records*.

For example: let’s define an arrayClassList, which stores information about the students in the table above.

Student ClassList[5]   or   ClassList : **Array** [1..5] of **Student**

An element in this type of array is referenced as follows:
ArrayName[index].fieldName

ClassList[0].Gender == F; ClassList[4].DOB == 29 Feb 1995; ClassList[3].DOB.Month == Dec

13.1.2.3 Stack

A stack is a linear list in which items are added and removed from the same end (head). Items being added and removed from the same end means that the last item to be added will be the first to be removed. As such stacks are also called last in, first out (LIFO) lists as they use the LIFO principle. (FILO)
Stack operations are:

a) **Stack(S):** creates an empty stack named S
b) **Push(S,x):** inserts the element x to the top of the stack S. Inserting an element into a stack that is full leads to a situation known as overflow. (START == NULL)
c) **Pop(S):** removes an element from the top of the stack S. Removing an element from an empty stack leads to a situation known as underflow. (START == HEAD)
d) **IsEmpty(S):** checks whether the stack S is empty. Returns true if it is empty and false otherwise.

Application of stacks:

i. ***Converting a Decimal Number to Binary***

Initially we push the binary digit formed into the stack, instead of printing it directly. After the entire digit has been converted into the binary form, we pop one digit at a time from the stack and print it. Therefore we get the decimal number converted into its proper binary form.

ii. ***Evaluating Arithmetic Expressions***

Arithmetic expressions are usually written in the form$(a + b)$, where the operator is in between the operands. This is known as the infix notation. This notation poses problems for more complicated expressions. For example:

$$a \times b + c = \begin{cases} (a \times b) + c? \\ a \times (b + c)? \end{cases}$$

$$2 \times 6 + 5 = \begin{cases} (2 \times 6) + 5 = 17? \\ \quad or \\ 2 \times (6 + 5) = 22? \end{cases}$$

Infix notation requires the use of order of precedence of operators and parentheses making it complicated and difficult for computers to evaluate expressions in this form.

To ease evaluation of arithmetic expressions, other notations are used: the prefix and postfix notations.

In ***prefix notation*** also called polish notation, operations are written before the operands. For example the expression $(a \times b) + c$ will be written in prefix notation as

$$(a \times b) + c = (\times ab) + b$$
$$= + \times abc$$

In ***postfix notation*** also called reverse polish notation, operands come before operators. For example

$(a \times b) + c = (ab \times) + c = ab \times c +$ $\quad$ $(2 \times 3) + 1 = 23 \times 1 +$

Also,

$a \times (b + c) = a \times (bc +) = abc + \times$ $\quad$ $2 \times (3 + 1) = 231 + \times$

Prefix and *postfix notation*s do not require parentheses or precedence rules. Calculators use postfix notation to evaluate arithmetic expressions. A simple way to understand is by using the following expression: **Pre** A **In**B **Pos.**

To convert from infix to prefix or postfix, priorities are assigned to operators as follows: $priority(\times) = priority(/) > priority(+) = priority\,(-)$ and the following rules are used.

- When an operand lies between two operators, it associates with the operator that has higher priority.
- When an operand lies between operators of same priority, it associates with the operator on the left.

Example 1: Let's convert the expression P: $A + B \times C/D$ to postfix.

Considering the priority rules given above, we can rewrite P as follows:

P:$A + ((B \times C)/D)$

Stack[empty]

Output: [ ]

Read A push it into stack ➔stack[A] output: [A]

Read + ➔stack[+] output: [A]

Read (➔stack[+(] output: [ ]

Read (➔stack[+((] output: [ ]

Read B ➔stack[+((] output: [AB]

Read * ➔stack[+((*] output: [AB]

Read C ➔stack [+((*] output: [ABC]
Read) ➔stack [+(] output: [$ABC \times$] pop until you meet an opening)
Read / ➔stack [+(/] output: [$ABC \times$]
Read D ➔stack [+(/] output: [$ABC \times D$]
Read) ➔ stack [+] output: [$ABC \times D/$]
➔stack[ ] output: [$ABC \times D/+$]

$A + ((B \times C)/D)$in postfix notation is $ABC \times D/+$

Example 2: Let's evaluate the expression $234 \times 5 + -$ written in postfix notation.

Read 2 ➔ stack [2]
Read 3 ➔ stack [$2, 3$]
Read 4 ➔ stack [$2, 3, 4$]
Read $\times$ ➔ stack [$2, 3 \times 4 = 12$]= [$2, 12$]
Read 5 ➔ stack [$2, 12, 5$]
Read $+$ ➔stack [$2, 12 + 5 = 17$] = [$2, 17$]
Read $-$ ➔ stack [$2 - 17 = -15$] = [-15]

The result of the expression $234 \times 5 + -$ is -15

Exercise 1: Convert the following to postfix notation and evaluate them.

i) $2 - 3 \times 4 + 5$ $(34 \times 2 - 5+)$
ii) $(2 - 3) \times (4 + 5)$ $(23 - 45 +\times)$
iii) $2 \times 3/(2 - 1) + 5 \times 3$ (23x

Exercise 2: Evaluate the following expressions

i) $623 + -382/+ \times 2\hat{}3 +$
ii) $12, 7, 3, -, /, 2, 1, 5, +, \times, +$

13.1.2.4 Queues

A queue is a linear list in which items are inserted at one end (tail/rear) and removed from the other end (head/front) such the first item added to the queue is the first to be removed. Queues are also called FIFO lists as they use the FIFO principle. (LILO)

Queue operations include:

- **Create(Q):** creates a queue Q.
- **Enqueue(Q,x):** inserts the element x to the tail of the queue Q.
- **Dequeue(Q):** removes the first element from the head of the queue Q
- **Top():** checks the next element to be removed (do not remove)
- **IsEmpty():** checks whether the queue is empty. Returns true if it is empty and false otherwise.

Application of queues:

a. Spooling

It is a method used to place input and output on a fast access storage device, such as a disk, so that slow peripheral devices do not hold up the processor. For example, print spooler stores output to be printed in a queue while waiting for the user's program to finish creating the output. The spooler then sends the stored output to the printer at the proper speed. If new output arrives for printing, it is added to the queue.

b. Scheduling

It is the process of determining which of the jobs in memory will be executed by the CPU. When a job is created, it is added to a queue, the ready queue. Any other job that arrives is added to the tail of the queue. A scheduling algorithm like round-robin is then used to allocate the jobs one by one, to the CPU for execution.

Remark *A queue in which elements can be added and removed from any end is known as a double ended queue (Deque).*

13.1.2.5 Linked Lists

A linked list is a linear list in which each element contains a pointer to the next element in the list. Each element in a linked list is known as a node and consists of two parts: the data item and the pointer to the next element.

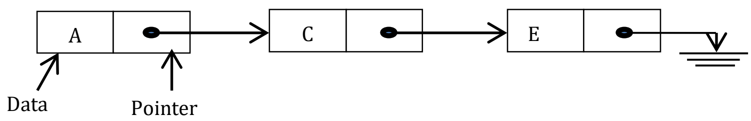


Figure 13.1: *A linked list that stores characters*

Linked lists make it possible to insert items in the middle of the list without moving other items to make room.

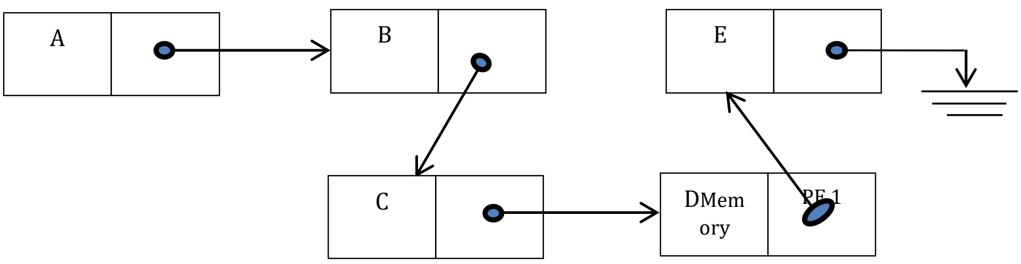


Figure 13.2: *Linked list with elements C and D inserted.*

NB: *In a double linked list, every node has two pointers: one for its predecessor and another for its successor in the list.*

13.1.2.6 Binary Tree

A binary tree is a finite set of nodes that is either empty or consists of a root and two disjoint binary trees called the left sub-tree and the right sub-tree. In other words, it is a non-linear linked list where each node may point to two other nodes. Binary trees are a very efficient way of storing items that must be searched for and retrieved quickly. Suppose, for example, that you want to store the following names in a computer using a binary tree:

Jones, Stephen, Agbor, William, Bate, Peter, Terence

The names can be arranged into a binary tree by using the following two-step procedure:
1. Use the first name on the list as the root of the tree.
2. To find where to put each subsequent name, start at the root of the tree. If the name you are dealing with precedes the root name alphabetically, follow the left pointer; otherwise follow the right pointer. Proceed in this way until you come to an empty pointer; attach the name to it.

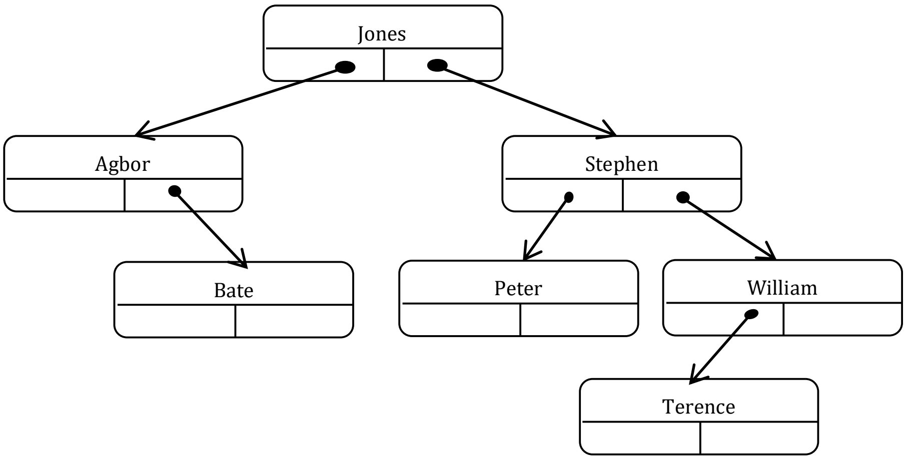


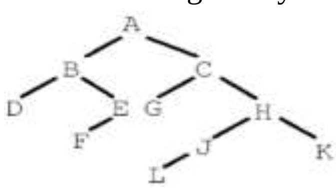
Figure 13.3: *A binary tree that stores names*

The topmost node in the tree is known as the *root node*. Each node in a binary tree may have at most two *children* or *child nodes*. A node that has a child is called the *child's parent node* (or ancestor node, or superior). A node has at most one parent except the root node that has no parent. Nodes that have the same parent are called *siblings*. Every node in a tree can be seen as the root node of the sub-tree rooted at that node.
Nodes at the bottom most level of the tree are called leaf nodes**.** Since there are at the bottom most level, they will not have any children.

Tree traversal is the process of systematically visiting all the nodes in a tree and performing some computation. When describing a traversal strategy, we need not concern ourselves with what that computation is. There are two methods for traversing a tree: breadth first traversal and depth first traversal.

a. Breadth First Traversal

In a breadth first traversal all of the nodes on a given level are visited and then all of the nodes on the next level are visited, usually in a left to right fashion. It is also called level order traversal.
Consider the following binary tree.



A level order traversal of this tree gives: ABCDEGHFJKL

b. *Depth First Traversal*

In depth first traversal, the left subtree is traversed, the right subtree is traversed and the root is visited. There are three different depth first traversal techniques; preorder, in order and post order traversals. What distinguishes them is the order in which the subtrees and the root are visited.

- ✓ Preorder Traversal:
  - Visit the root
  - Traverse left subtree
  - Traverse right subtree

A preorder of the above tree gives: ABDEFCGHJLK

- ✓ Inorder Traversal:
  - Traverse left subtree
  - Visit the root
  - Traverse right subtree

An in order traversal of the tree above gives: In order traversal: DBFEAGCLJHK

- ✓ Postorder Traversal:
  - Traverse left subtree
  - Traverse right subtree
  - Visit the root

A post order traversal of the tree above give: DFEBGLJKHCA

13.1.2.7 Hash Table

A hash table is an array in which data is stored at specific locations designated by a hash function. The hash function transforms the value of a record key into an index that corresponds to a location for storing the record. In other words, it maps the set of input data to a set of integers. Each element to be stored in the array has a unique key that is mapped by the hash function to a numeric value that represents an index in the array.
For example,

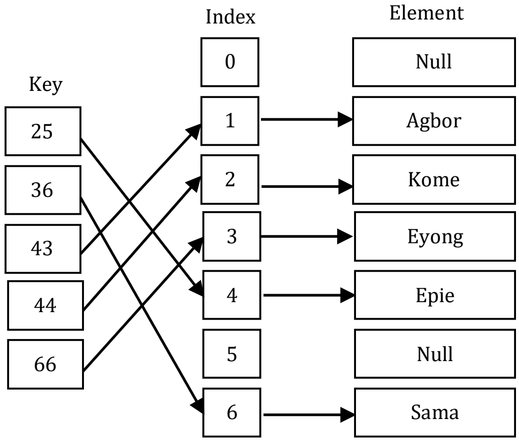


Figure 13.4: *A hash table*

The above hash table stores the names Agbor, Epie, Sama, Kome and Eyong using the hash function *h(k) = k mod tablesize*, where k, the key of each string is the sum of its letters' positions in the alphabet. This means that if an element *p* has key *k*, then *p* will be stored at position *h(k)* in the table.

Keys:

Agbor = 1+7+2+15+18 = 43
Epie = 5+16+9+5 = 25
Sama = 19+1+13+1 = 34
Kome =11+15+13+5 = 44
Eyong = 5+25+15+14+7 = 66

Applying the function *h* to each key we get:

Agbor = 43 mod 7 = 1
Epie = 25 mod 7 = 4
Sama = 34 mod 7 = 6
Kome = 44 mod 7 = 2
Eyong = 66 mod 7 = 3

Hash table operations are:

- **Search**: compute $h(k)$ and see if an element exists
- **Insert**: compute $h(k)$ and place element in the resulting position
- **Delete**: compute $h(k)$ and remove element in that position

Remarks! The size of the array should be preferably a prime number.

With hash tables, there always exists the possibility that two data elements will hash to the same integer value. This situation is known as collision. Two methods to solve collision are separate chaining and probing (closed hashing).

13.2 Algorithms

An algorithm is a well-defined set of step-by-step instructions for solving a problem in a finite amount of time. A set of instructions is not an algorithm if there is no definite stopping place, or if the instructions are too vague to be followed clearly.

A good algorithm:
(i) *Should be explicit (i.e. clear and obvious)*
(ii) *Should be precise (i.e. exact and accurate)*
(iii) *Should be unambiguous (i.e. no doubts about what to do/ only one way of interpreting the instructions)*
(iv) *Should be effective (i.e. produce good results)*
(v) *Should be finite (i.e. have a definite stopping place)*

13.2.1 Representation of Algorithms

There are different ways for representing algorithms such as pseudo code, flow chart and structured English.

13.2.1.1 Pseudo code

A pseudo code is an outline of a computer program, written in a mixture of a programming language and English. Writing pseudo code is one of the best ways to represent an algorithm as it allows the programmer to concentrate on how the program works while ignoring the details of the language.
In a pseudo code, some terms are commonly used to represent the various actions. For example, for inputting data the terms may be (INPUT, GET, READ), for outputting data (OUTPUT, PRINT, DISPLAY), for calculations (COMPUTE, CALCULATE), for incrementing (INCREMENT), in addition words like ADD, SUBTRACT, INITIALIZE are used for addition, subtraction, and initialization, respectively.
For example, here is a pseudo code for an algorithm that reads two numbers, computes and displays their sum:

```
Begin
        Get A
        Get B
         Result = A + B
        Print Result
End
```

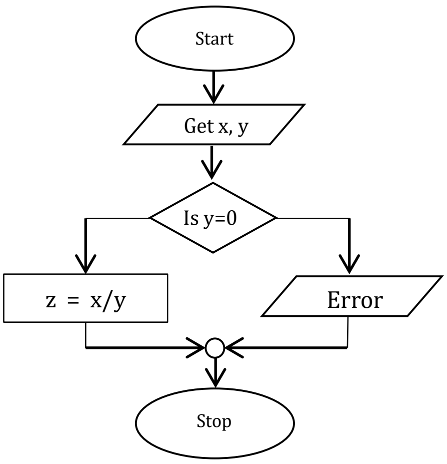
13.2.1.2 Flowchart

A flow chart is a diagram that uses graphic symbols to describe the nature and flow of steps in an algorithm. Each step in a flowchart is followed by an arrow that indicates which step to go next. The following symbols are used in flow charting:

Name	Symbol	Usage
Start or Stop	Start/Stop	The beginning and end points in the sequence.
Process	Process	An instruction or a command.
Decision	Decision	A decision, either yes or no. For example, a decision based on temperature that turns a central heating system on or off.
Input or output	Input/Output	An input is data received by a computer. An output is a signal or data sent from a computer.
Connector		A jump from one point in the sequence to another.
Direction of flow		Connects the symbols. The arrow indicates direction.

Table 13.1: *Flow chart symbols and their corresponding usages*

Example 1: flow chart for an algorithm that divides two numbers x and y.



13.2.2 Variables, Constants and Literals

A variable is an object in a program whose value can be modified during the execution of the program. In the above flow chart, x, y and z are variables.

A constant is an object whose value cannot be modified in the course of the algorithm or program. A constant is given a value that remains the same all through the program.

Variables and constants are characterized by:

- **an identifier**: which is the name of the object

- **a value**: which is the content of the object
- **a type**: which defines the domain in which the object gets its value

A literal is anything (numbers or text) that is usually written within double quotes. For example, “Enter a number”, “The result is”.

13.2.3 Basic Instructions

Three basic instructions used in an algorithm are input, output and assignment instructions:

- An *input instruction* allows information to be typed from the keyboard.
  Example: read (a, b), get (number)
- An *output instruction* allows:
  - ✓ display of information on the screen
  - ✓ printing of information on paper

  Example: write “a is greater than b” or print “b is greater than a”
- The *assignment statement* allows a value to be assigned to a variable. A variable can be assigned the content of another variable, a constant, a literal, an arithmetic or Boolean expression. The symbol used is ←.

Examples: $z \leftarrow x/y$, $sum \leftarrow a + b$, $total \leftarrow total + 1$, $pi \leftarrow 3.14$
In an assignment statement, the value to the right is assigned to the variable in the left.
In the case of $sum \leftarrow a + b$, “$a + b$” is calculated and the result is assigned (kept) in the variable sum.
For $total \leftarrow total + 1$, “$total + 1$” is calculatedand the result is assigned to the variable total, meaning that the value of total has been increased by 1.
E.g. Let $total = 3$
$total \leftarrow total + 1 \Rightarrow total = 3 + 1 = 4$.

Activity: Complete the table below.

$Command$	x	y	z
$x \leftarrow 2$		$-$	$-$
$x \leftarrow x + 1$		$-$	$-$
$y \leftarrow 1$			$-$
$z \leftarrow x + y$			
$y \leftarrow z + x$			

13.2.4 Control Structures

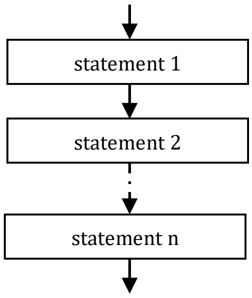
The logic of a program may not always be a linear sequence of statements to be executed in that order. The logic of a program may require execution of a statement based on a decision. It may repetitively execute a set of statements unless or until some condition is met. Control structures specify the statements to be executed and their order of execution.

13.2.4.1 Sequential Control

A sequence control structure executes a set of instructions one after the other from the first to the last in the order they are given.

Syntax: ***begin***
statement 1
statement 2
…
statement n
end

```
Begin
  get a
  get b
  c← a + b
  print "Sum =",
c
End
```

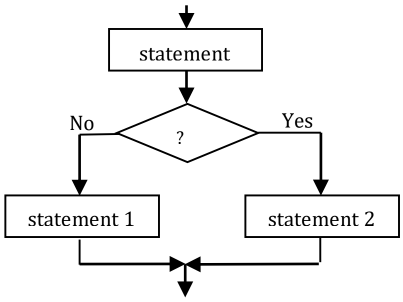


13.2.4.2 Selection Control

A selection control structure (condition control structure) chooses the instruction or instructions to be executed based on the validity of a certain condition. Examples are IF and CASE statements.

a) The IF Statement

Syntax: ***if*** *condition* ***then***
statement 1
else
statement 2



Explanation *Condition* is a Boolean expression meaning that it can take only one of two values true or false. The *condition* is evaluated, if it is true, *Action3* is executed. If it is false, *Action 2* is executed.

Note that actions 2 and 3 could be a block statements.

Example:

```
Get a, b
if (a = 0) then
  Print "Error"
else
  Print b/a
```

It is possible to nest many selection structures.

Syntax: ***if*** *condition1* ***then***
if *condition2* ***then***
statement 1
else
statement 2
else
statement 3

```
get a, b
if a <> 0 then
if b <> 0 then
print b/a
else
print "Answer is 0"
else
print "Error: division by 0"
end
```

Explanation If *condition1* is true, we move to *condition2*. If *condition2* is true, then *statement 1* is executed otherwise, *statement 2* is executed. If condition 1 is false, instruction 3 is executed. Instruction1 or instruction 2 will be executed if and only if condition 1 is true.

b) The CASE Statement

Syntax: ***case** variable **of***

case 1: statement 1
case 2: statement 2
...
case n: statement n
end case

Explanation The value of variable is evaluated, if it matches with case 1, instruction 1 is executed. If it matches with case 2, instruction 2 is executed and so on. CASE is a multiple selection structure. It is used when an important number of choices are to be considered depending on the value of a variable.

Example:

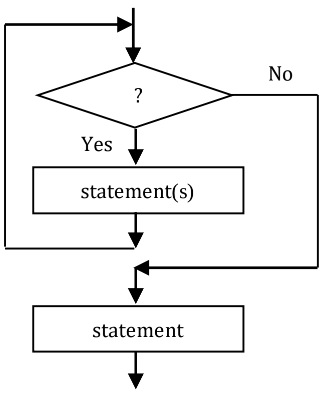
```
Get a, b
Get op
caseop of
   1: print a+b
   2: print a*b
   3: print a/b
   4: print a-b
   -
```

13.2.4.3 Repetition Control

The repetition (iteration) control structure executes a statement or group of statements many times until a certain condition is reached. Repetition structures define the order of operations and the number of repetitions. They are also called loops. Examples are, the WHILE, REPEAT and FOR loops.

a. The WHILE Loop

Syntax: ***while** condition **do***
statement(s);
end while



Explanation The *condition* is evaluated, if it is true statement(s) is/are executed. Instruction(s) is/are executed as long as condition remains true. When the condition becomes false, the loop stops.

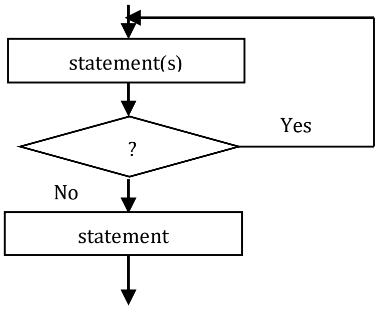
The condition for the loop to stop comprises of a variable called control or iteration variable whose value must change at the end of each execution of the loop. In the example above, the control variable is “i”.

Example:

```
Get n
i← 1
while (i<= n) do
print “this is a while loop”
i←i + 1
endwhile
```

b. The REPEAT Loop

Syntax: ***repeat***
statement(s);
***until** (condition)*



Explanation The statement(s) is(are) executed and the condition is evaluated. If it evaluates to false, the statement or set statements is/are executed again. If condition evaluates to true, the program exits the loop.

```
Get n
i← 1
repeat
print “This is a repeat loop”
i := i + 1
until(i<= n)
```

Remark! *The repeat loop must be executed at least once as the condition is evaluated only at the end of the loop.*

c. The FOR Loop

Syntax: ***for** var ←low_limit**to** hi_limit**do***
statement(s);
end for
Or
for**var ←hi_limit**downto**low_limit**do
statement(s);
end for

Explanation var (variable) is given a value *low_limit* or *hi_limit* depending on the loop, which is automatically incremented or decremented (by 1) after each iteration of the loop. The loop stops when *low_limit* becomes greater than *hi_limit*. In both cases, if *hi_limit* is less than *low_limit*, the loop body is not executed at all.

Example:

```
Get n
For i← 1 to ndo
print "this is a for loop"
end For
                Or
Get n
For i← n downto 1 do
print "this is a for loop"
end For
```

Exercises:

1) Write an algorithm that reads a number n and returns the first n numbers. n should be a whole number greater than 0.
2) Write an algorithm that reads a number n and returns the sum of the first n numbers.
3) Write an algorithm to calculate the area of a circle.
4) Write an algorithm to solve a linear equation
5) Write an algorithm that reads a person's name and sex, and returns the message "good morning Mr. name" if the person is a man and "Good morning Mrs. Name" for a woman.

13.2.5 Recursion

Some problems are recursive in nature. This means that the solution to such problems involves the repeated application of the solution to its own values until a certain condition is reached. Algorithms for such problems are known as recursive algorithms.
A recursive algorithm is an algorithm that calls (invokes) itself during its execution. Examples are the factorial function and the sum function.
Recursion can be defined as the calling of a procedure by itself, creating a new copy of the procedure.

a. Factorial Function

Factorial is defined as:

$1! = 1$

$2! = 2 \times 1 = 2$

$3! = 3 \times 2 \times 1 = 6$

$4! = 4 \times 3 \times 2 \times 1 = 24$

$5! = 5 \times 4 \times 3 \times 2 \times 1 = 120$

...

$n! = n \times (n-1) \times (n-2) \times \ldots \times 2 \times 1$

By studying the above equations closely, we see that the factorial of any number n can be calculated by multiplying the number by the factorial of the preceding number.
We therefore have:

$1! = 1$

$2! = 2 \times 1!$

$3! = 3 \times 2!$

$4! = 4 \times 3!$

$5! = 5 \times 4!$

$\cdots$

$n! = n \times (n-1)!$

Factorial is defined recursively as follows:

$$fact\ (n)\begin{cases} 1 & if\ n = 1 \text{ base case} \\ n \times fact(n-1) & if n > 1 \end{cases}$$

Get n
If $n =\ 0$ **or** $n = 1$ **then**
    $fact$ ← 1
else
  $fact$ ← $n \times fact(n-1)$
return fact

Remark 1! = 1 is known as the base case. Every recursive problem must always have some base case which can be solved without recursion. For cases that are to be solved recursively, the recursive call must always make progress towards the base case.

b. The Sum Function

The sum function is a function that calculates the sum of the first n integers. For example we want to calculate the sum of the first 5 integers 1, 2, 3, 4 and 5. Their sum is calculated as follows:

Sum(5) = 1+2+3+4+5

We can see that for any number *n*, the sum *sum (n)*, is the number *n* plus the sum of the previous numbers.

The sum function can therefore be defined recursively as:

$sum(1)\ =\ 1$
$sum(2)\ =\ 2\ +\ sum(1)$
$sum(3)\ =\ 3\ +\ sum(2)$
$sum(4)\ =\ 4\ +\ sum(3)$
...
$sum(n) = n + sum(n-1)$

The base case is $n = 1$ which gives $sum(1) = 1$

get$n$
sum ← 0
if $n = 1$ **then**
    sum ← 1
else
  sum ← n + $sum(n-1)$
return sum

c. The Fibonacci Series

The Fibonacci series is a series in which each number is the sum of the two previous numbers in the series. E.g. 0, 1, 1, 2, 3, 5, 8, 13, 21…

The Fibonacci series is defined as follows:

$fib(0)\ =\ 0$
$fib(1)\ =\ 1$
$fib(n)\ =\ fib(n-1)\ +\ fib(n-2)$

$fib(0)$and$fib(1)$ are the base cases.

Get n
fib ← 0
if $n = 0$ **then**
    fib ← 0
else
  if $n\ =\ 1$ **then**
    $fib\ = 1$
  else
    fib ← fib$(n-1) + fib(n-2)$
return fib

Exercise: Write a recursive algorithm to find the greatest common divisor (gcd) of two numbers.

13.2.6 Sort Algorithms

Sorting is a programming technique which is used to arrange a list of pre-stored data in an ascending or descending order according to a preset criterion. There are lots of useful sorting methods. For example: insertion sort, bubble sort, selection sort, quick sort, merge sort and heap sort algorithms.

a. Bubble Sort

An example of a computer algorithm is *bubble sort*. This is a simple algorithm used for taking a list of jumbled up numbers and putting them into the correct order. The algorithm runs as follows:

1. Look at the first number in the list.
2. Compare the current number with the next number.
3. Is the next number smaller than the current number? If so, swap the two numbers around. If not, do not swap.
4. Move to the next number along in the list and make this the current number.
5. Repeat from step 2 until the last number in the list has been reached.
6. If any numbers were swapped, repeat again from step 1.
7. If the end of the list is reached without any swaps being made, then the list is ordered and the algorithm can stop.

(i) Bubble sort example

This algorithm could be used to sort the following list:

3, 2, 4, 1, 5

The first loop of the algorithm would produce:

- **3**, **2**, 4, 1, 5 (2<3 so the two values are swapped)
- 2, **3**, **4**, 1, 5 (3<4 so the two values are **not** swapped)
- 2, 3, **4**, **1**, 5 (1<4 so the two values are swapped)
- 2, 3, 1, **4**, **5** (4<5 so the two values are **not** swapped)
- 2, 3, 1, 4, 5 (**First pass completed**)

Values were swapped so the algorithm needs to run again. The second loop of the algorithm would start with the final list and run again as follows:

- **2**, **3**, 1, 4, 5 (2<3 so the values are **not** swapped)
- 2, **3**, **1**, 4, 5 (1<3 so the values are swapped)
- 2, 1, **3**, **4**, 5 (3<4 so the values are **not** swapped)
- 2, 1, 3, **4**, **5** (4<5 so the values are **not** swapped)
- 2, 1, 3, 4, 5 (**Second pass completed**)
- **2**, **1**, 3, 4, 5 (1<2 so the values are swapped)
- 1, **2**, **3**, 4, 5 (2<3 so the values are **not** swapped)
- 1, 2, **3**, **4**, 5 (3<4 so the values are **not** swapped)

- 1, 2, 3, **4**, **5** (4<5 so the values are **not** swapped)
- 1, 2, 3, 4, 5 (**Third pass completed**)

Values were swapped so the algorithm needs to run again. This time there will be no swaps as the values are in order: 1, 2, 3, 4, 5

b. Selection Sort

Selection ort is a simple sorting algorithm. This sorting algorithm is a in-place comparison based algorithm in which the list is divided into two parts, sorted part at left end and unsorted part at right end. Initially sorted part is empty and unsorted part is entire list.

Smallest element is selected from the unsorted array and swapped with the leftmost element and that element becomes part of sorted array. This process continues moving unsorted array boundary by one element to the right.

This algorithm is not suitable for large data sets as its average and worst case complexity are of $O(n^2)$ where n are no. of items.

How selection sort works?

We take the below depicted array for our example.

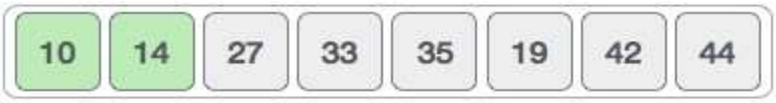
For the first position in the sorted list, the whole list is scanned sequentially. The first position where 14 is stored presently, we search the whole list and find that 10 is the lowest value.

So we replace 14 with 10. After one iteration 10, which happens to be the minimum value in the list, appears in the first position of sorted list.

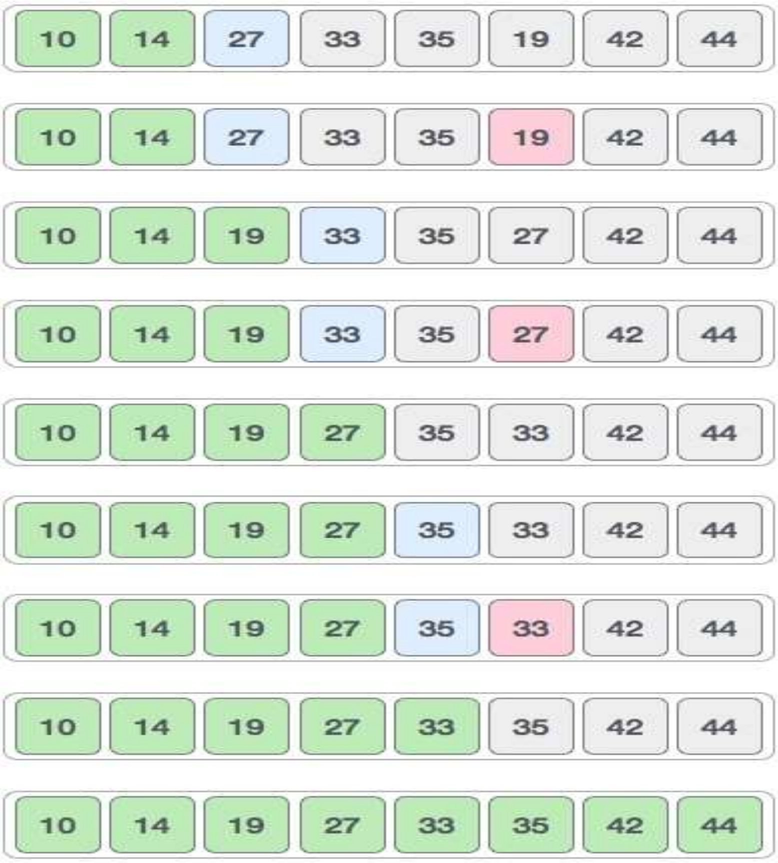
For the second position, where 33 is residing, we start scanning the rest of the list in linear manner.

We find that 14 is the second lowest value in the list and it should appear at the second place. We swap these values.

After two iterations, two least values are positioned at the the beginning in the sorted manner.



The same process is applied on the rest of the items in the array. We shall see a pictorial depiction of the entire sorting process −



c. Insertion Sort

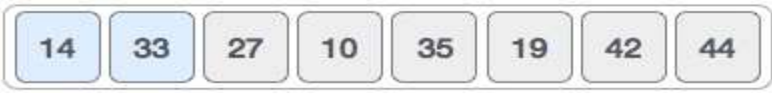
This is a in-place comparison based sorting algorithm. Here, a sub-list is maintained which is always sorted. For example, the lower part of an array is maintained to be sorted. An element which is to be *'insert'ed* in this sorted sub-list, has to find its appropriate place and insert it there. Hence the name **insertion sort**.

The array is searched sequentially and unsorted items are moved and inserted into sorted sub-list (in the same array). This algorithm is not suitable for large data sets as its average and worst case complexity are of O (n^2) where n are no. of items.

How insertion sort works?

We take an unsorted array for our example.

Insertion sort compares the first two elements.



It finds that both 14 and 33 are already in ascending order. For now, 14 is in sorted sub-list.

Insertion sort moves ahead and compares 33 with 27.

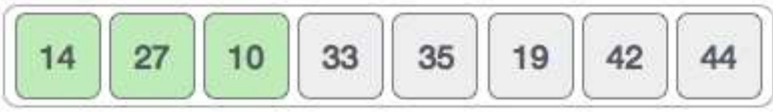
And finds that 33 is not in correct position.

It swaps 33 with 27. Also it checks with all the elements of sorted sublist. Here we see that sorted sub-list has only one element 14 and 27 is greater than 14. Hence sorted sub-list remain sorted after swapping.

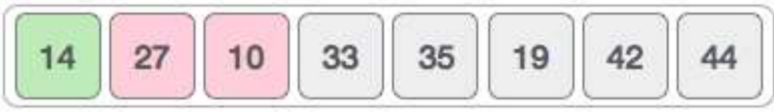
By now we have 14 and 27 in the sorted sub-list. Next it compares 33 with 10,.

These values are not in sorted order.

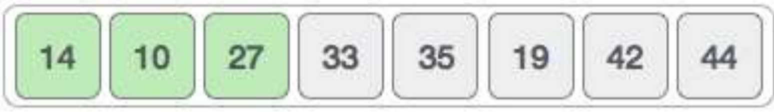
So we swap them.



But swapping makes 27 and 10 unsorted.



So we swap them too.



Again we find 14 and 10 in unsorted order.

And we swap them. By the end of third iteration we have a sorted sub-list of 3 items.

This process goes until all the unsorted values are covered in sorted sub-list

13.2.7 Search Algorithms

A search algorithm is a method of locating a specific item in a larger collection of data. They can be used to search for items within an array or list. Common search algorithms are sequential search and binary search.

a. Linear Search

Linear search is a very simple search algorithm. In this type of search, a sequential search is made over all items one by one. Every items is checked and if a match founds then that particular item is returned otherwise search continues till the end of the data collection.

b. Binary Search

Binary search is a fast search algorithm with run-time complexity of O(log n). This search algorithm works on the principle of divide and conquer. For this algorithm to work properly the data collection should be in sorted form.

Binary search searches a particular item by comparing the middle most item of the collection. If match occurs then index of item is returned. If middle item is greater than item then item is searched in sub-array to the right of the middle item otherwise item is search in sub-array to the

left of the middle item. This process continues on sub-array as well until the size of subarray reduces to zero.

c. Interpolation Search

Interpolation search is an improved variant of binary search. This search algorithm works on the probing position of required value. For this algorithm to work properly the data collection should be in sorted form and equally distributed.

Binary search has huge advantage of time complexity over linear search. Linear search has worst-case complexity of O (n) whereas binary search has O (log n).

There are cases where the location of target data may be known in advance. For example, in case of telephone directory, if we want to search telephone number of Morphius. Here, linear search and even binary search will seem slow as we can directly jump to memory space where names start from 'M' are stored.

13.3 Programming

Programming is the activity of writing computer programs. A computer program is a set of instructions that will be followed by a computer to perform a computation. These instructions are made up of statements written in some languages specially designed for this purpose. These languages are called programming languages.
In other words, a program is an algorithm expressed in a programming language.

13.3.1 Programming Languages

A programming language is a set of predefined words, symbols and rules that are used to write computer programs. Programming languages are grouped into low-level and high-level languages.

13.3.1.1 Low-Level Languages

A low-level language is a language whose instruction set reflects the processor architecture. An instruction set is the set of bit patterns or binary codes for the machine operations that a processor has been designed to perform. Low-level languages include machine language and assembly language.

a. Machine Language

Machine language is the computer's language. It is the language the computer understands. Machine language instructions are written in binary (a series of 0s and 1s), and are directly executable by the computer. Each machine language statement corresponds to one machine action. Machine language is the first generation of programming languages. For example a short (3 instruction) program might look like this:

```
0111 0001:      0000 1111
1001 1011:      0001 1010
```

1110 0001:   0011 1110

b. Assembly Language

Assembly language is a low-level language consisting of mnemonic codes and symbolic addresses corresponding to machine language instructions. Assembly language is the second generation of programming languages. For example:

```
LOAD   R0   Number1          Load number1 in register 0
LOAD   R1   Number2          Load Number2 in register 1
ADD    R2   R0   R1    Add register 0 and register 1 and keep result in register 2
```

a) Advantages of assembly language

- It is easier to write and understand when compared to machine language.
- It can produce small program sizes
- It can produce very fast code as it allows low-level access to hardware features

b) Disadvantages of assembly language

- Programs are not as easy to write and understand when compared to high level languages.
- Programs are tied to specific computer hardware and can't be reused on another kind of computer.
- Writing programs is very time consuming, tedious, and error-prone.

13.3.1.2 High-Level Languages

High-level languages are closer to human language. They allow programmers to write programs without having to understand the inner workings of the computer. One high-level language statement will generally be translated into several low-level language statements. They are the third generation of programming languages. Examples are C, BASIC (**B**eginner's **A**ll-purpose **S**ymbolic **I**nstruction **C**ode), Pascal, Java, FORTRAN (**For**mula **Tran**slator) and COBOL (**Co**mmon **B**usiness-**O**riented **L**anguage). Below is a small code for adding two numbers in Pascal and C.

Pascal

```
program addition;
usescrt;
var number1, number2, sum: integer;
begin
  read(Number1);
  read(Number2);
  sum := Number1 + Number2;
  write(sum);
end.
```

C

```
#include <stdio.h>
intmain()
{       int  number1,  number2,
sum;
  scanf("%d", &number1);
  scanf("%d", &number2);
  sum = number1 + number2;
  printf("%d", sum);
  return 0;
}
```

a) Advantages of high-level languages

- easy to understand and write programs as they are user oriented
- they have built in libraries to perform routine tasks
- programs can be ported to multiple hardware setups from same code

b) Disadvantages of high-level languages

- programs may be slower than second generation languages
- may produce larger program files for same functionality as second generation languages.
- may not allow for low level hardware access

13.3.2 Language Translators

To run a program on a computer, the program needs to be translated into the machine language of the computer on which it will run. A language translator is a computer program that translates program instructions from one programming language to another without loss of original meaning. There are three types of language translators: *compiler, interpreter and assembler*.

13.3.2.1 Assembler

An assembler translates assembly language into machine code (also known as source code).Assembly language is a low-level language written in mnemonics that closely reflects the operations of theCPU.

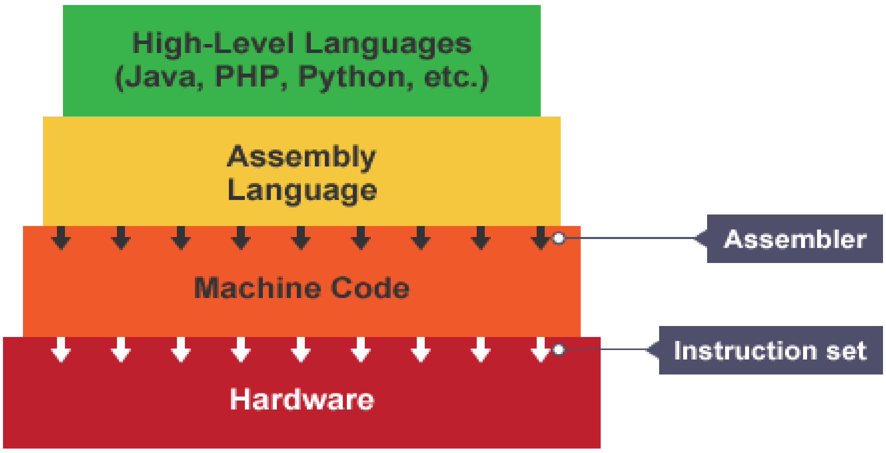


Figure 13.5: *Instruction flow from High level language to Hardware*

13.3.2.2 Compiler

A compiler translates the entire high-level program into a machine language program. The high-level language program is called *source program* or *source code* and the generated machine language program is called *object program* or *object code*. This process is called *compilation*. Some compilers convert high-level language into assembly language, then an assembler is used to create the finished object code.

a) Advantages of a Compiler

- Fast in execution
- The object code produced by a compiler can be distributed or executed without having to have the compiler present.
- The object program can be used whenever required without the need of recompilation.

b) Disadvantages of a Compiler

- Debugging a program is much harder. Therefore not so good at finding errors

- When an error is found, the whole program has to be re-compiled

13.3.2.3 Interpreter

An interpreter *translates code into machine code, instruction by instruction* - the CPUexecutes each instruction before the interpreter moves on to translate the next instruction. Interpreted code will show an error as soon as it hits a problem, so it is easier to debugthan compiled code.
An interpreter does not create an independent final set of source code - source code is created each time it runs. Interpreted code is slower to execute than compiled code.
Interpreted languages include JavaScript, PHP, Python and Ruby. Interpreted languages are also called *Scripting languages*. These are ideal for using within dynamic web applications. They are used for client-side and server-side coding, as they are small programs that are executed within the browser.

a) Advantages of an Interpreter

- It is good at locating errors in programs
- Debugging is easier since the interpreter stops when it encounters an error.
- If an error is corrected, there is no need to retranslate the whole program

b) Disadvantages of an Interpreter

- It is slow as interpretation and execution is done line by line.
- Translation has to be done every time the program is to be executed since no object code is produced.
- For the program to run, the interpreter must be present

13.3.3 Virtual machines

A process virtual machine is a programming environment that allows a program written for one type of machine to run on other types of machine without any changes being necessary. For example, the Java Virtual Machine allows Java bytecode produced on a Microsoft Windows OS to run on a Unix OS without any changes.
A system virtual machine is a software application run by the host operating system of a computer which emulates a second operating system. This permits the installation and execution of software applications on the virtual machine (VM) as if they were being installed and run on a separate computer.

For example, it is possible to run a Microsoft Windows Virtual Machine on an Apple Mac - allowing software applications that only work on the Microsoft Windows OS to be run via the VM, which in turn is run on the Mac.

13.3.4 IDE

An integrated development environment (IDE) is an application used to create software. An IDE can often support different languages.

IDEs have a number of different tools and functions that assist a developer in the creation of software.

13.3.4.1 Code editor

The environment where the user can write code is called the shell. The code editor is a text edit area that allows developers to write, edit and save a document of code. It has features that assist with the writing and editing of code. These include:

- ***Auto-completion*** (or *code completion*). This is designed to save time while writing code. As you start to type the first part of a function, it suggests or completes the function and any arguments or variables.
- ***Bracket matching***. This is used for languages that use pairs of brackets to mark out blocks of code. It allows the code to be read and understood more quickly. If you forget to close a bracket while writing, colored sections may help you to detect missing brackets.
- ***Syntax checks***. This recognizes incorrect use of syntax and highlights any errors.

IDEs have a runtime environment. This means that you can execute the program one step at a time. This is useful to test that the code is working line by line before creating the final complete program.

13.3.4.2 Other IDE tools

The IDE also includes several tools to automate and speed up processes.

- ***Translator***. This compiles or interprets the code.
- ***Auto documentation***. This explains the function and purpose of the code, e.g. by noting the modules and variables used, and its expected behavior, and collates this into a text file that can be used by other developers to understand how and why the code was created.
- ***Libraries***. These provide functions that are not included in the core part of the programming language. These functions can be imported and used at the start of the program code. For example, in Python the Turtle Graphics library provides access to some simple drawing and graphics tools.
- ***Build automation***. These tools save time by automatically doing the processes that would otherwise be done by hand. These could include testing or compiling. These tools are extremely useful when a program has many thousands of lines of code. They improve the quality of the software, minimizing bad software builds as well as saving time and money.
- ***Debugger***. This is a program within the IDE that is used to detect errors. If the debugger detects errors, it may suggest what the type of error is and what line it is on.

13.3.5 Program Errors and Correction

13.3.5.1 Syntax Errors

Syntax is the set of rules that specify how the symbols of a language can be put together to form meaningful statements. In other words, syntax defines the structure of legal statements in a language. A syntax error is an error in a program that occurs due to the non-respect of the syntax rules of the language used. A syntax error will cause a compiler/interpreter to stop trying to generate machine code and will not create an executable. However, a compiler will usually not stop at the first error it encounters but will attempt to continue checking the syntax of a program right to the last line. For example, a misspelled key word, a missing punctuation mark or the incorrect use of an operator is a syntax error.

13.3.5.2 Semantic Errors

Semantics specify the meaning of a well-formed program. A semantic error occurs when you write a program that works, but does not do what you intend it to do. Compilation and interpretation do not detect semantic errors. Semantic or logic errors are detected from wrong results. Something may be syntactically correct but semantically incorrect.

13.3.5.3 Run-time Errors

A run-time error is an error that occurs during program execution. For example, a run-time error may occur if division by 0 is attempted. A run-time error may cause the program to stop execution, or it may be handled by an error-trapping routine.

13.3.5.4 Debugging

An error in a computer program is known as a bug. Debugging is the process of detecting and removing bugs. Syntax errors and semantic errors are bugs. A debugger is the software tool used for this purpose.

13.3.6 Programming Paradigms

A programming paradigm (or technique) is a fundamental style of computer programming. It describes a programming language's approach to solving a problem. Paradigms differ in the concepts and abstractions used to represent the elements of a program and the steps that compose a computation. High level languages can be classified under four different paradigms: procedural, functional, object-oriented and declarative paradigms.

13.3.6.1 Procedural Paradigm

In the procedural/imperative paradigm, a program is a collection of statements and procedures that affect data. Here, a program can be seen as an active agent that manipulates passive objects(variables).These objects are passive because they cannot initiate an action by themselves, but can only receive actions from active agents. The focus in procedural programming is to write good functions and procedures.
Examples of imperative languages are Pascal, C, Ada, FORTRAN, and COBOL.

13.3.6.2 Object Oriented Paradigm

The object-oriented paradigm presents a program as a collection of classes for interacting objects. Unlike in imperative programming, object-oriented programming deals with active objects instead of passive objects. These objects are active because the actions to be performed on the objects are included in them (the objects). The objects need only to receive the appropriate stimulus from outside to perform one of the actions.
Examples of object-oriented languages are C++, Java, Visual Basic and Smalltalk.
Some important concepts related to object oriented programming (OOP) are: class, object, abstraction, encapsulation, inheritance and polymorphism.

a. Class

A class is a description of an object or a real life concept. Classes are templates for creating objects, providing initial values for instance variables (attributes) and the bodies for methods. All objects generated from the same class share the same methods, but contain separate copies of the instance variables. New objects can be created from a class by applying the *new* operator to the name of the class.

```
Class Person {
    private:
        charname[20];
        charsex;
        datebirthdate;
public:
        updateInfo();
        returnAge();
  };
```

The name of the class is person. It has four instance variables (attributes) *name, sex* and *age,* and three methods: *updateInfo(),*and *returnAge()*. These methods act on the instance variables when invoked.

b. Object

An object is an instance of a class. An object consists of a collection of *attributes*, representing the state of the object, and a collection of *methods*, representing the behaviour that the object is capable of performing. Attributes are sometimes referred to as the fields of an object. The methods are routines that are capable of accessing and manipulating the values of the attributes of the object. Objects interact with each other by sending messages. When a message is sent to an object, the corresponding method of the object is executed.

c. Abstraction

Data abstraction refers to, providing only essential information to the outside world and hiding their background details, i.e., to represent the needed information in program without presenting the details.

Data abstraction is a programming (and design) technique that relies on the separation of interface and implementation.

Let's take one real life example of a TV, which you can turn on and off, change the channel, adjust the volume, and add external components such as speakers, VCRs, and DVD players, BUT you do not know its internal details, that is, you do not know how it receives signals over the air or through a cable, how it translates them, and finally displays them on the screen.

Thus, we can say a television clearly separates its internal implementation from its external interface and you can play with its interfaces like the power button, channel changer, and volume control without having zero knowledge of its internals.

d. Encapsulation

Encapsulation is the process of combining together the attributes and methods of a class into a single abstract data type with a public interface and a private implementation. The goal of encapsulation is to protect the implementation from the users of the object. It ensures that all access to the internal representation of the object pass through the class methods, which acts as an "interface" to the object. This is done by making properties and methods “private.”

e. Inheritance

Inheritance is the derivation of one class from another so that the attributes and methods of the derived class are part of the definition of the initial class. The initial class is often called the base class, parent class or superclass while the derived class is often referred to as the child class or sub-class. The subclass usually contains all the attributes and methods of the superclass plus some of its own.

For example: the class Student inheriting from the class person will have all the attributes and methods of Person plus the following:

```
Class Student extends Person {
  private
    intadmNumber;
    intlevel;
  public
    changeLevel(intnewLevel) {
        level=newLevel;
         }
}
```

Inheritance is usually represented using an inheritance diagram.

For **example**: if the classes Student and Teacher are derived from the superclass Person, this is represented as:

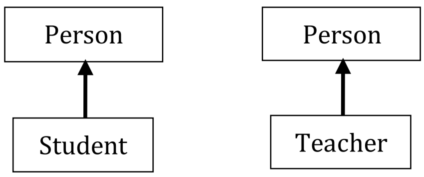


Figure 13.6: *Inheritance diagram*

There are two types of Inheritance: *multiple inheritance* and *multilevel inheritance*.

i) **Multiple inheritance** is when a derived class inherits features from more than one superclass. It is illustrated as follows:

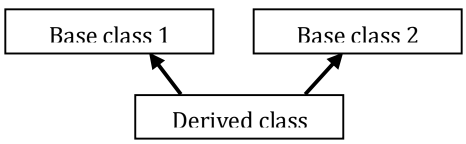


Figure 13.7: *Multiple Inheritance diagram*

ii) **Multi-level inheritance** is when a class inherits from a class which is itself inherited from another class. It is illustrated as follows:

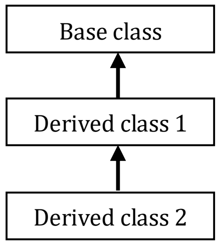


Figure 13.8: *Multilevel Inheritance diagram*

f. Polymorphism:

Polymorphism is the use of different methods, each with the same name, which are associated with different object types. In other words, it is the ability for different objects to respond to the same message in different, class-specific ways. Polymorphic methods are used which have one name but different implementations for different classes.

Assume you have a "shape" superclass. This class has a method called "area" which returns the area of the shape. Polymorphism allows you to make subclasses like "circle," "square," and "triangle" which inherit the "area" method, but each subclass would return the correct value even though they have different formulas to calculate their areas.

13.3.6.3 Functional Paradigm

In functional (applicative) programming, a program is a collection of function definitions. Lambda calculus forms the basis of almost all functional programming languages. Lambda calculus is the use of lambda expressions to define functions. A lambda expression is a formula that defines a function. For example: $f(x) = x + 2$

Examples of functional languages are Haskell, LISP, ML and Scheme.

Sum function in Haskell

$$add :: (Int, Int) \rightarrow Int$$
$$add\ (x, y) = x + y$$

Quick sort in Haskell

$f :: [a] \rightarrow [a]$

$f[\ \ ] = [\,]$

$f(x{:}xs) = fys ++ [x] ++ fzs$

Where

$ys = [a \mid a \leftarrow xs, a \leq x]$

$zs = [b \mid b \leftarrow xs, b > x]$

13.3.6.4 Declarative Paradigm

In declarative (logic) programming, a program is a collection of facts and rules involving relational expressions. Declarative paradigm uses the principle of logical reasoning to answer queries. It is based on formal logic defined by Greek mathematicians and later developed into first-order predicate calculus. The point of logic programming is to bring the style of mathematical logic to computer programming. An example of a declarative programming language is PROLOG (programming logic).

These facts are given:
human (John)
mortal (human)

The user asks the following question:
?-mortal (John)

The program answers yes.

13.3.7 C Programming

C is a high-level programming language developed by Dennis Ritchie and Brian Kernighan at Bell Labs in the mid-1970s. Although originally designed as a systems programming language, C has proved to be a powerful and flexible language that can be used for a variety of applications, from business programs to engineering.
The first major program written in C was the UNIX operating system, and for many years C was considered to be inextricably linked with UNIX. Now, however, C is an important language independent of UNIX.
C is a particularly popular language for personal computer programmers because it is relatively small - it requires less memory than other languages.

13.3.7.1 Basic Structure of a C Program

A C program is made up of the following components:
- ✓ Processor directive
- ✓ Declaration of variables
- ✓ Declaration of functions
- ✓ Function main()
- ✓ Definition of functions

a. The Pre-processor Directive
A pre-processor directive is a statement that begins with the # symbol. It instructs the compiler to include C pre-processor such as header files and symbolic constants before compiling the C program. There are two categories of pre-processor directives: the include directive and the define directive.

✓ **The Include Directive**
An “include directive” is used to include a header file. A header file is a library file that contains declarations for a special group of functions that can be used in the main body of the program. A header file must be included at the beginning of a program, if its functions are to be used in the program. The extension of a header file is ".h". A program may contain many header files. The general syntax to include a header file at the beginning of a program is:

```
#include <name of header file>
```

For example:

```
#include <stdio.h>
#include <math.h>
```

The file "stdio.h" contains standard input and output functions that are used to get input (scanf) and print output (printf). The file "math.h" contains math functions used for mathematic calculations.

✓ **The Define Directive**

The "define directive" is used to define a symbolic constant which is assigned a value that will remain constant during the execution of the program. Its general syntax is:

`#define` identifier expression

Where

- Identifier specifies the macro name to which a value will be assigned
- Expression specifies the value that is assigned to the identifier. It may be a constant value, a string or an arithmetic expression.

For example:

```
#define  Pi  3.14
#define  CityLimbe
```

b. Declaration of Variables

A variable is a memory location reserved to contain a value that may change during the execution of the program. Variables are used for storing input data or values generated as result of processing. Variables are characterized by their name, type and scope.

✓ **Variable Names**

A variable's name or identifier in C can be anything from a single letter to a word. However, it must begin with a letter or the underscore character but the other characters in the name can be chosen from the following sets:

a ..z	(any letter from a to z)
A .. Z	(any letter from A to Z)
0 .. 9	(any digit from 0 to 9)
_	(the underscore character)

Examples of valid variable identifiers are: x, total, area_of_circle, x1, _a and aX.

Remark! *C is case sensitive, so the identifiers sum and SUM are different. Same for total and Total. All key words must be in lower case.*

✓ **Variable Types**

Every variable has a type. The type of the variable specifies what sort of data will be stored in it. The type of a variable is specified during its declaration. To declare a variable in C, one writes the *type* followed by the variable *name or identifier.*

The general syntax for declaring a variable is:

[Type] [Name of variable]

For example:

```
int   age;
float   area;
int   sum = 0;
inti,j=2;
```

- *int* specifies that the variables age and sum are integers while *float* specifies that the variable area is a floating point (real or decimal) number.
- *int sum=0* indicates that variables can be assigned values when they are declared. This is called initialization. Therefore, one can initialize a variable at the time of declaration.
- *i* and *j* are integers with j *initialized* at 2.

Variable declaration serves two purposes:

- It gives the compiler precise information about the amount of memory that will have to be given over to a variable when a program is finally run and what sort of operations will have to be used on it.
- It provides the compiler with a list of the variables in a convenient place so that it can cross check names and types for any errors.

Some basic C types are:

	Type	**Description**
1.	char	a single ASCII character
2.	int	standard integers (usually 32 bits)
3.	long int (long)	long integers
4.	float	standard floating point or real numbers
5.	long float (double)	long floating point numbers
6.	unsigned int	positive short integers
7.	unsigned float	positive standard floating point numbers
8.	unsigned long	positive long integers
9.	unsigned double	positive long floating point numbers

Table 13.2: *C identifier types and their respective descriptions*

☺ **C fun!**
The bank said they couldn't FLOAT our loan because we were LONG on debt and SHORT of funds. Our application was UNSIGNED, so the whole deal was VOID. Can you believe it? The bank rejected us in C-speak!

✓ **Scope of a Variable**

The scope of a variable describes where in the program the variable can be legally used. Based on its scope, a variable can be global or local.
A global variable is a variable that can be recognized anywhere in the program. Global variables are declared before the function main().
A local variable is a variable that has meaning only within a particular function or other program unit. Local variables can be declared anywhere following the opening brace ({) of a block. The name of a local variable can be used in another block elsewhere in the program, where it will refer to an entirely different variable.

c. The Function main( )

The function main() indicates the beginning of the actual C program. It is the point at which execution of program is started. When a C program is executed, the execution control goes directly to the function main(). Every C program must have a function main(). The general syntax is:

```
int main()
  {declaration of local variables;
  program statements;
  return 0;
  }
```

- The keyword "*int*" is used before the main () function to indicate the type of the value that is returned by the main() function. By definition, a function may accept no, one or more inputs and returns no or a single value. 'int' means that the program will return an integer value after its execution. The word "void" can be used in the place of "int" to indicate that the program will not return any value.
  It is good practice to always return a value because the operating system uses the return value to determine whether the program has been executed successfully or not.

- The body of the main() function must be enclosed in braces (or curly brackets { }). These braces are called delimiters. The left brace indicates the start of the body of the function whereas the matching right brace indicates the end of the body of the function. Braces are also used to indicate the beginning and ending of block (compound) statements.

Remark! *The use of '**void**' or '**int**' before the function main() is optional. If it is not specified, the compiler assumes it is **int**.*

- The body of the function main () is made up of program statements which represent instructions to be executed by the computer. An instruction may be an input/output statement, an arithmetic statement, a control statement, a simple assignment statement or any other statement.

A basic C program looks like this:

```
#include   <header.h>
#define   symbolic constant
Global declarations
void   main()
{ local declarations
  program instructions;
 }
```

```
#include   <stdio.h>
int  main()
  {
 printf("Welcome  to  the  world  of
C!");
 return 0;
  }
```

☺ **C fun!**

C programmers know that the real reason the Roman empire fell was that, because they had no number zero, they had no way to successfully return from C programs.

13.3.7.2 Input/Output Statements

a. The *scanf* Function

scanf (print-formatted) is used to interpret characters input to the computer and to store the interpretation in the specified variable(s).

Example: `scanf("%d", &x);`

This statement reads a decimal integer from the keyboard and stores the value in the memory address of the variable x.

In the statement,

i. `%d`is a *conversion specifier* and specifies that the variable to be read is of type integer. Other conversion specifiers used in C are `%f`for floating point numbers, `%c`for characters and `%s`for strings.

ii. `&`gives the *address* of something in memory. That is, it generates a *pointer* to the object. The arguments to scanf*must* be pointers (addresses), hence the need for &.

b. The *printf* Function

printf(print-formatted) is used to display information on the screen.

Example: `printf("The radius is %f cm",x);`

in the statement,

i) "The radius is `%f  cm`" is the control string. The text in between the double quotes will be displayed except the conversion specifier %f.

ii) `x` is the variable to be printed. The value of x will be printed where the conversion specifier is placed in the control string.

Many conversion specifications can be used in a control string. In this case, the programmer has to ensure that both the number of conversion specifications and the number of variables to be printed are the same and, of the correct type specified.

Example: `printf("The sum of %d and %d is %d", x, y, x+y);`

If x=2 and y=3, then what will be displayed on the screen is: The sum of 2 and 3 is 5

c. Common Conversion Specifiers

Character	Form of output
c	Character
d	Decimal integer
f	Normal floating point
s	String
e	Scientific notation floating point

Table 13.3: *Common conversion specifiers and their corresponding descriptions*

Examples:

```
scanf("%s", &Name);
printf("Name: %s", Name);
scanf("%d %d",&x,&y);
printf("The sum of %d and %d = %d", x, y, z);
```

d. Character Input/Output

getchar and *putchar* are used for the input and output of single characters respectively.
getchar() returns an int which is either*EOF* (indicating end-of-file) or the next character in the standard input stream
putchar(c) puts the character c on the standard output stream

```
#include<stdio.h>
main()
{ charch;
ch=getchar();
printf("%c",ch);
 }
```

13.3.7.3 Operators and Expressions

An operator is something which takes one or more values and does something useful with those values to produce a result. C has different types of operators which can be arithmetic, relational, logical, assignment operators etc.
An expression is simply the name for any string of operators, variables and numbers.

a. Arithmetic Operators

Arithmetic operators are used for arithmetic calculations.

	Operator	Description
1.	+	Addition

2.	−	Subtraction
3.	∗	Multiplication
4.	/	Division
5.	%	Remainder after division(modulo arithmetic)

Table 13.4: *Arithmetic operators and their descriptions*

b. Relational Operators

Relational operators are used for comparisons. Expressions that use these operators produce a true or false value when they are evaluated.

	Operator	Description	Example
1.	==	Equal to	If a == 2 and b== 5; a == b evaluates to FALSE
2.	<	Less than	a < b evaluates to TRUE
3.	>	Greater than	a > 5 evaluates to FALSE
4.	<=	Less than or equal to	b <= 5 evaluates to TRUE
5.	>=	Greater or equal to	a >= 5 evaluates to FALSE
6.	!=	Not equal to	a != b evaluates to TRUE

Table 13.5: *Relational operators, their descriptions and examples of usage*

c. Logical Operators

Logical operators are used to combine logical values.

	Operator	Description	Example
1.	&&	Logical AND	(a>b) && (a>C)
2.	||	Logical OR	If ((a ==0) || (a ==1)), printf("a!=1");
3.	!	Logical NOT (Negation)	If found == 1 (true), !(found) == 0 (false)
4.	&	Bitwise AND	4 & 5 = 0100 & 0101 = 0100 = 4
5.	|	Bitwise OR	4 | 5 = 0100 | 0101 = 0101 = 5
6.	^	Bitwise XOR	3 ^ 9 = 0011 ^ 1001 = 1010 = 10

Table 13.6: *Logical operators, their descriptions and examples of usage*

Remark! *Bitwise operators allow manipulation of the actual bits held in each byte of a variable. Other bitwise operators are:*

- ✓ Right shift (>>) – binary division by 2
  E.g. if $a\ ==\ 0010111$ then $a{>>}=\ 0001011.$
  Also, $a{>>}2$ = 0000101
- ✓ Left shift (<<) – binary multiplication by 2
  If $a\ ==\ 0010111$ then $a{<<}=\ 0010110.$
  Also, $a{<<}$ 2 = 1011100

- ✓ One's complement (~)
  If $a\ ==\ 0010111$ then $a \sim\ =\ 1101000$

d. Assignment Operators

Assignment is the process of storing a value in a variable. The assignment operator is the equal sign (=).

For example:

$$pi\ = 3.14$$
$$sum = sum + 1$$

The variable $pi$is assigned the value 3.14 and the variable sum is incremented by 1.

C has different types of assignment operators.

	Operator	Description	Example
1.	=	assign	c = 2
2.	+=	Assign with add	$a\ = a\ + b\ \equiv\ a$ +=b
3.	-=	Assign with subtract	$a\ = a - b\ \equiv\ a$ -=b
4.	*=	Assign with multiply	$a\ = a * b\ \equiv\ a$ -=b
5.	/=	Assign with divide	$a\ = a/b\ \equiv\ a$ /=b
6.	%=	Assign with remainder	$a\ = a$ % $b\ \equiv\ a$ %= b
7.	>>=	Assign with right shift	a>>= b
8.	<<=	Assign with left shift	a<<= b
9.	&=	Assign with bitwise AND	$a\ = a$ & $b\ \equiv\ a$ &=b
10.	|=	Assign with bitwise OR	$a\ = a \mid b\ \equiv\ a$ |=b
11.	^=	Assign with bitwise XOR	$a\ = a$ ^ $b\ \equiv\ a$ ^=b

Table 13.7: *Assignment operators, their descriptions and examples of usage*

e. Increment and Decrement Operators

Increment and decrement operators give a shorthand method of adding and subtracting 1 from an object respectively.

++	Increment	a++ $\equiv a\ =\ a + 1$
--	Decrement	b-- $\equiv\ b = b - 1$

These operators can be prefix or postfix. With the prefix form the variable is changed before the value of the expression in which it appears is evaluated, and with the postfix form the variable is modified afterwards.

$b = 3;$ $a = b{+}{+} + 6;$ /* a = 9, b = 4 */	$b = 3;$ $a = {+}{+}b + 6;$ /* a = 10, b = 4 */

13.3.7.4 Control Flow Statements

Control flow statements make it possible to make decisions, to perform tasks repeatedly or to jump from one section of code to another. C control flow statements are If statement, switch statement, while statement, do … while statement, for statement and jump statements.

a. The IF Statement

The IF statement is a decision-making statement that is used to evaluate an expression and then take one of two possible actions depending on the validity of the expression. The IF statement has two forms.

Syntax: ***if****(expression)*
statement(s);
and

```
if (condition)
{
  statement(s);
}
else
 {
  statement(s)
 }
```

In the first form, if the expression specified in the if statement evaluates to true, the statements inside the if-block are executed and then the control gets transferred to the statement immediately after the if-block.
In the second form, the else part is required only if a certain sequence of instructions needs to be executed if the expression evaluates to false.

```
scanf("%d", &b);
if (b>0)
 printf("Number %d is positive",
b);
else
```

- ✓ The expression is always enclosed within brackets.
- ✓ If statement(s) is a single statement, the curly brackets can be omitted as in the example above though it is good practice to always enclose them with curly brackets.

Sometimes we may want to evaluate more than one thing in an expression. This can be done in C by using the logic operators AND and OR.

For example:

- ✓ **if** (a==0 || b==1)

This expression is TRUE if a is 0 **OR** b is 1. It is also true if $a = 0$ **and** $b = 1$.

- ✓ **if** (a == 1 && b == 2)

This expression is TRUE if a is 0 **AND** b is 1. If one is NOT TRUE, the whole expression is evaluated to FALSE. In other words, a must equal 0 and b equal 2, for the expression to be TRUE.

b. Nested IF

It is also possible to embed or to nest IF statements one within the other. Nesting is useful in situations where one of several different courses of action need to be selected.

Example:

```
#include<stdio.h>
int main()
{  int a, b;
  scanf("%d %d ", &a, &b);
   if(a>b)
     {
    printf("%d is greater.", a);
       }
    else
     if(a==b)
     {
     printf("%d and %d are the same: ", a, b);
     }
     else
     {
     printf("%d is greater.", b);
     }
  return 0;
     }
```

c. SWITCH Statement

A switch statement is used for multiple way selections that will branch into different code segments based on the value of a variable.

Syntax: ***switch****(variable)*

{

case*value1 : code segment1; break;*

case*value2 : code segment2; break;*

...

case*valueN : code segment N; break;*
default*: statement(s);*
}

If the value of the variable equals ‘value1’, code segment1 is executed. Otherwise, if it equals value2, code segment2 is executed and so on.
The statement break is used after every case in order to prevent execution from continuing into the code segment of the next case without even checking its value.
For example, supposing a switch statement has five cases and the value of the third case matches the value of expression. If no break statement were present at the end of the third case, all the cases after case 3 would also get executed along with case 3. If break is present only the required case is selected and executed; after which the control gets transferred to the next statement immediately after the switch statement. There is no break after default because after the default case the control will either way get transferred to the next statement immediately after switch.

```
/* C program to demonstrate the working of switch...case statement */
/* C Program to create a simple calculator for addition, subtraction,
multiplication and division */

# include <stdio.h>
int main(){
char o;
float num1,num2;
printf("Select an operator either + or - or * or / \n");
scanf("%c",&o);
printf("Enter two operands: ");
scanf("%f%f",&num1,&num2);
switch(o){
case'+':
printf("%.1f + %.1f = %.1f",num1, num2, num1+num2);
break;
case'-':
printf("%.1f - %.1f = %.1f",num1, num2, num1-num2);
break;
case'*':
printf("%.1f * %.1f = %.1f",num1, num2, num1*num2);
break;
case'/':
printf("%.1f / %.1f = %.1f",num1, num2, num1/num2);
break;
default:
/* If operator is other than +, -, * or /, error message is shown */
printf("Error! operator is not correct");
break;
}
return0;
}
```

Remark! If you need to select among a large group of values, a switch statement will run much faster than a set of nested IFs.
The switch differs from the IF in that switch can only test for equality, whereas IF can evaluate any type of Boolean expression.

d. WHILE Statement

The While statement or while loop is an iteration statement. Iteration statements are used to execute a particular set of instructions repeatedly until a particular condition is met or for a fixed number of iterations.

Syntax: ***while*** *(condition)*
```
{
  statements;
}
```

The *statement or statements* are only executed if the *expression* is true (non-zero). After every execution of the *statements*, the *expression* is evaluated again and the process repeats if it is true.

Example:
```
inti=1;
while(i<=10)
  {
 printf("%d", i);
 i++;
  }
```
This code will loop 10 times writing the numbers 1 to 10.

Example: A program that counts the number of blank spaces in a line of text.

```
#include <stdio.h>
int main()
{ charch; short count = 0;
 printf("Type in a line of text\n");
 while((ch = getchar()) != '\n')
      {
if(ch == ' ')
count++;
      }
printf("Number of spaces = %d\n", count);
return 0;
}
```

e. DO… WHILE Statement

The do-while statement evaluates the condition at the end of the loop after executing the block of statements at least once. If the condition is true the loop continues, else it terminates after the first iteration.

Syntax*:* ***do** {*
statements to be executed;
} ***while****(expression*);

Remark!

- ✓ Pay attention to the semicolon which ends the do-while statement.
- ✓ The difference between while and do-while is that the while loop is an entry-controlled loop - it tests the condition at the beginning of the loop and will not execute even once if the condition is false, whereas the do-while loop is an exit-controlled loop - it tests the condition at the end of the loop after completing the first iteration.

Example: A program to print the sum of the digits in a number.

```
#include<stdio.h>
int main()
{ int n, a,sum=0;
 printf("Enter a number:");
 scanf("%d", &n);
 do{
          a =n%10;
sum=sum+a;
 n=n/10;
}  while(n>0);
  printf("Sum of the digits = %d",sum);
return 0;
}
```

A practical use of the do-while loop is in an interactive menu-driven program where the menu is presented at least once and then depending upon the choice of the user, the menu is displayed again or the session is terminated. Consider the same example that we saw in switch-case. Without using an iteration statement like do-while, the user can choose any option from the menu only once. Moreover, if a wrong choice is entered by mistake the user doesn't have the option of entering his choice again. Both these faults can be corrected by using the do-while loop.

f. FOR Statement

The FOR statement or the FOR loop repeatedly executes a set of instructions that comprise the body of the loop until a particular condition is satisfied.

Syntax: ***for***(*initialization; termination; increment/decrement*;)
{

statements to be executed;
}

- ✓ The initialization expression initializes the looping index which controls the looping action. The initialization expression is executed only once, when the loop begins.
- ✓ The termination expression represents a condition that must be true for the loop to continue execution.
- ✓ The increment/decrement expression is executed after very iteration to update the value of the looping index.

Example: A program to multiply two numbers by successive additions.

```
#include <stdio.h>
int main()
{ int a, b, i, prod=0;
 scanf("%d",&a);
 scanf("%d",&b);

 for (i=1;i<=b;i++)
  prod=prod+a;

 printf("%d",prod);
 return 0;
}
```

13.3.7.5 Arrays

An array is a collection of data items of the same type that are given a single name (identifier) and distinguished by numbers (subscripts). In simple terms, an array is a list of elements of the same type. When an array is created (declared), its size (dimension) and the type of its elements are specified. Arrays can be one dimensional or multidimensional.

a. One-Dimensional Arrays

One-dimensional arrays are declared using a single index (subscript).
For example:

int A[10], char Name[10], intvect[20]

int A[10] creates an array called "A" of 10 integers. The elements of array A are referenced as: A[0], A[1], [2], … A[8], A[9]

Where A[0] is the first element in the array, A[1] the second element, A[2] the third element and A[9] the last element.

Values can be assigned to the array as if each element were a separate variable.
A[0] =3; A[1] =1; A[2] =0; … A[9] =-5

A loop can as well be used.
For example:

```
int main()
{  inti, n=10, A[n]
i=0;
while(i<=n)
 {
 printf("Enter element at position %d", i);
 scanf("%d", &A[i]);
 i++;
 }
 return 0;
}
```

To display the individual elements of the array A, we can proceed as follows.

```
printf("%d", A[0]);//prints the first element of the array.
printf("%d", A[1]); //prints the second element of the array.
printf("%d", A[2]);//prints the third element of the array.
…
printf("%d", A[n-1]);//prints the last element of the array.
```

Using a loop to display the elements of an array.

```
for(i=0; i<=n; i++)
 printf("%d", A[i]);
```

b. Two-Dimensional Arrays

Two dimensional arrays are declared using two indices (subscripts).
For example:

int A[3][2], int B[5][3], int M[3][4]

int M[3][4] declares a 3x4 array where 3 is the number of rows and 4 the number of columns. The elements of the array M can be visually represented as a table as follows:

M[0][0]	M[0][1]	M[0][2]	M[0][3]
M[1][0]	M[1][1]	M[1][2]	M[1][3]
M[2][0]	M[2][1]	M[2][2]	M[2][3]

The elements of a 2-dimensional array can be assigned using nested loops or as separate variables.

- ✓ As separate variables

```
M[0][0] = 2;  //element on row 1, column 1
M[1][2] = 5; //element on row 2, column 3
```

- ✓ Using nested loops

```
int main( )
{  inti, j, n=3, m=4;
for(i=0; i<n;  i++)
for(j=0; j<m; j++)
 {
printf("M[%d][%d] = ", i, j);
scanf("%d %d", &M[i][j]);
 }
return 0;
}
```

The elements of a 2-dimensional array can be displayed using nested loops or as separate variables.
Using nested loops, we have

```
int main( )
{ inti, j, n=3, m=4;
for(i=0; i<n;  i++)
for(j=0; j<m; j++)
printf("M[%d] [%d]=%d ", i, j, M[i][j]);
 return 0;
}
```

13.3.7.6 User-Defined Functions

A user defined function is a subprogram written by the user to perform a specific task. User defined functions have to be declared and defined. They are declared before the function main but, may be defined (written) at declaration or after the function main().

a. Function Declaration

A function declaration (also called function prototype) is simply a statement which specifies the function's type, name and parameter or argument list. Function names follow the same rules as variable names. The general syntax for declaring a function is:

Type functionName(parameter list, if any)

- ✓ Type refers to the type of the value the function returns.
- ✓ Parameters or arguments refer to the function's inputs. A function may take no parameter, one or more parameters and returns no or a single value. If a function returns no value, it is of type "void". If the function takes in any parameters, their types and names are specified (declared) within the brackets.

Example:

`intsquare(int n)` a function which takes an integer$n$ and returns its square (n^2).

`intpower(int b, int n)` a function which takes two integers b and n and returns b^n

`char getchar()` a function that takes no parameter and returns a character.

Remark! Parameter names are optional in a function prototype, so for the above prototypes we could have written:

```
int square(int)
int power(int, int)
```

b. Function Definition

Function definition consists of writing the actual statements of the function. A function definition has the form:

Type functionName(parameter list, if any)
{
local declarations; declaration of local variables
statements; function statements
}

For example:

```
Float areaCircle(int r)
{ float area;
  area = 3.14*r*r;

  return (area);
 }
```

```
int square(int r)
{ intsqr;
  sqr=r*r;
  return (sqr);
  }
```

A function that has been defined can be called within the function main()or within any other function.

For example:

```
#include<stdio.h>
Float areaCircle(int) //prototype

int main()
  {    float area; int rad;
  scanf("%d", &rad);
  area = areaCircle(rad);
  printf("The area is %f", area);
  return 0;
  }
```

```
#include<stdio.h>
int square(int) //prototype

int main()
  {    int a, b;
  scanf("%d", &a);
  b = square(a);
  printf("The square of %d is %d", a, b)
  return 0;
  }
```

Remarks!

- ✓ If the type of a function is not specified, C will always assume that the function is of type int.
- ✓ Notice that the function areaCircle was declared in the variables section of the function main(). This is because, if a function whose type is not int is not declared in the variables section of the function in which it is called, compilation errors will occur.

- ✓ The names used by a function for its parameters are local to the function, and are not visible to any other function: other routines can use the same names without conflict. This is also true of the variables declared within the function.
- ✓ When a function is called, parameters are passed to the function as inputs. In the above examples, rad and a are passed to the functions areaCircle and square respectively.
- ✓ These parameters which replace the formal parameters when a function is called are passed either by value or by reference. A parameter is passed by value when the value of the parameter is calculated and this is passed into the function. A parameter is passed by reference by using pointers. Pointers are addresses.

Drill Questions:

1.(i)(a) What is a programming language?
(b) Give one diffrerence between a low level language and a high level language?
(ii) With respect to programming, explain the following:
(a) Portability (b) Algorithm (c) Control Structure (**Q8ii, CGCE 2013**)

2.(i)(a) what is program control structure?
(b) Using either Pascal or C programming language write out the general form for the following control structures.
- Sequence - FOR statement - WHILE statement
(c) Draw a flowchart symbol for the choice/selection structure.
(ii)(a) What is a pseudo code?
(b) Write a psedo-code that requests marks for 5 subjects and calculates the average assuming all subjects have equal coefficients. *Note that average is given by (total marks)/5 in this case.*
(iii) Briefly describe the following system software stating the category of each.
(a) Compiler (b) Defragmenter (c) File compression (d) Interpreter
(iv)(a) Explain the term "programming paradigm"
(b) Define the following programming paradigms
- Procedural programming - Declarative programming (**CGCE 2015**)\

3.(i) Structured programming to a general methodology of writing good programs.
(a) What is programming?
(b) Give 4 properties of a good program.
(ii)(a) What is the main difference between a high level language (HLL) and Low level Language (LLL)?
(b) Explain what is meant by machine dependent. Which of these languages is machine dependent: HLL or LLL?
(c) Give one example of HLL.
(iii) It is usually necessary to compare different implementations of an algorithm to choose an optimal one. This can be done by considering their relative efficiencies.
(a) What is efficiency with regards to algorithm performance?
(b) Which two basic resources of a computer can be used to measure the efficiency of an algorithm?
(iv) Explain the following errors that can arise during programming:

(a) Syntax errors (b) Run-time errors (c) Logic errors (**CGCE 2016**)

4.(i)(a) What is meant by pseudocode?
(ii) Explain the following algorithm development concepts:
(a) Stepwise refinement (b) Top-down design
(iii) Briefly define and say what each of the following programming tools are used for:
(a) Compiler (b) Interpreter
(iv) Give the structure of the following loop constructs. Your structure should indicate the loop body(or statement). *The choice of your construct is shown below as an example.* Indicate for (a) and for (b) the loop conditions under which the loop continue to iterate.
(a) Repeat-until (or do-while) (b) WHILE (c) FOR (**Q4, CGCE 2014**)
Example:

```
IF condition THEN
  Statement
ELSE
 Statement.
```

CHAPTER 14: SYSTEM DEVELOPMENT AND PROJECT MANAGEMENT

14.1 Introduction

Most computer-based information systems are conceived, designed, and implemented using some form of systematic development process called software development process. In this process, end users and system analysts design systems based on an analysis of the information requirements of the information system to be built. The software development process is also called system development life cycle.

14.2 Software Development Life Cycle

SDLC is a structured step-by-step approach for creating and maintaining information systems. It consists of a number of stages that describe the activities involved in an information system development process. SDLC involves the following stages: *system study*, *system analysis*, *system design*, *development and testing*, *implementation*, and *maintenance.*

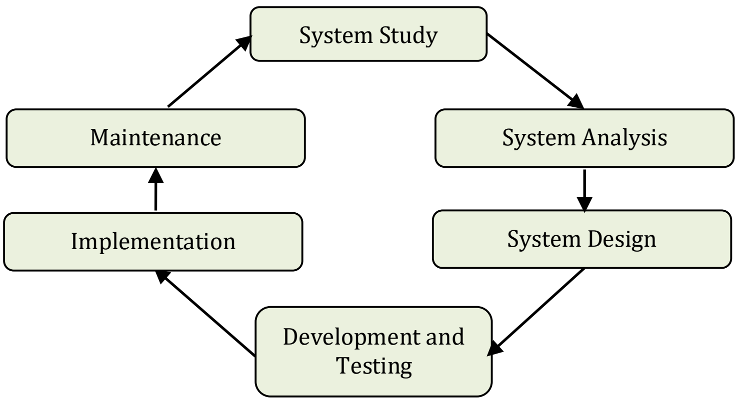


Figure 14.1:*System Development Life Cycle*

14.2.1 System Study

System study is a brief investigation of the system under consideration that gives a clear picture of what actually it is. During this phase, the system is evaluated and deficiencies are identified. This can be done by interviewing users of the system and consulting with support personnel. Main activities at this stage are:

- ✓ Determining whether a business problem or opportunity exists. i.e. identifying problems and opportunities.
  - A problem is a basic condition that is causing undesirable results
  - An opportunity is a basic condition that presents the potential for desirable results.
- ✓ Conducting a preliminary feasibility study to determine whether a new or improved information system is a feasible solution.
- ✓ Developing a project management plan and obtaining management approval.

14.2.2 System Analysis

Systems analysis is an in-depth study of end user information needs which produces functional requirements that are used as the basis for the design of a new information system. System analysis describes what a system should do to meet the information needs of users. It involves:

14.2.2.1 Analysis of the Old System

Analysis of present system involves:

a) Collecting factual data about the present system (questionnaires, interviews, observations, etc.)
b) Identifying how input, processing, storage and output are being accomplished.
c) Analysing how the present system uses resources (hardware, software and people) to convert input data into useful information
d) Understanding information flow within the system
e) Identifying problems with the system

14.2.2.2 Functional Requirements Analysis

Functional requirements explain what has to be done by identifying the necessary tasks, actions or activities that must be accomplished.
It involves:

a) Determining specific information needs
b) Determining the information processing capabilities required for each system activity (input, processing, output, storage, and control) to meet the needs. The goal here is to identify “what” should be done not “how” to do it.
c) Determining functional requirements (information requirements that are not tied to the hardware, software, and people resources that end users presently use or might use in the new system).

14.2.2.3 Feasibility Analysis

Feasibility analysis is a study which investigates the information needs of prospective users and determines the resource requirements, cost, benefits, and workability of a proposed project. Its goal is to evaluate alternative systems and propose the most feasible and desirable system for development.
Feasibility of a system can be evaluated in terms of four major categories: organisational feasibility, technical feasibility, economic feasibility and operational feasibility.

a. ***Organizational feasibility*** focuses on how well a proposed information system supports the objectives of the organization and its strategic plan for information systems.
b. ***Technical feasibility*** focuses on the reliability/capabilities of the hardware and software to meet the needs of the proposed system, and whether they can be acquired or developed in the required time.
c. ***Economic feasibility*** focuses on whether the tangible costs and benefits of the proposed system will exceed the costs of developing and operating it.
d. ***Operational feasibility*** focuses on the ability of the end users to operate, use, and support the proposed system.

The outcome of a feasibility analysis is a feasibility report which is presented to the user management for approval. It may be accepted or accepted with modifications or rejected.

14.2.2.4 Documenting System Analysis

The outcome of systems analysis is a system proposal or requirements specification document which describes what the new system should do without specifying how to do it. At the end of systems analysis phase, the system analyst produces a system proposal that will be used as basis for the design phase.

14.2.3 System Design

Systems design consists of design activities, which produce systems specifications satisfying the functional requirements developed in the systems analysis stage. While system analysis specifies what is to be done by the new system, system design describes how the system will accomplish what is to be done.
System design focuses on three main activities: user interface design, data design and process design.

14.2.3.1 User Interface Design

A user interface is a means of interaction between the user and the computer-based application. This activity focuses on designing how data will be introduced into the system and how the information generated will be retrieved. It produces detailed specifications for information products such as:

- Display screens
- Interactive user/computer dialogues
- Forms (on-screen forms for data input and output)
- Reports (on-screen and printed)

14.2.3.2 Data Design

Data design focuses on the design of the structure of data and files to be used by the proposed (new) system. It provides detailed descriptions of:

- Attributes (characteristics) of the entities about which the proposed system needs to maintain information.
- Relationships between these entities (E-R diagrams, data flow diagrams)
- Specific data elements (databases, files, records, etc.) that need to be maintained for each entity.
- Data dictionary
- Integrity rules (data validation and verification) that govern how each data element is specified and used in the system.

14.2.3.3 Process Design

Process design focuses on the design of software resources, that is, computer programs and of procedures needed by the proposed system. It concentrates on developing detailed specifications for the program modules that will have to be purchased as software packages or developed by custom programming. Process design produces:

- Detailed specification of algorithms (pseudo-codes, flow charts, etc.)

- Detailed specifications of the procedures needed to meet the user interface and data design specifications.
- Detailed specification of the database schema (E-R diagram, object diagrams)

The design stage is very important because it is the place where quality is fostered in software engineering. Design provides us with representations of software that can be assessed for quality. Design is the only way that we can accurately translate a customer's requirements into a finished software product or system.

14.2.4 Development and Testing

Once the design of the system is complete, it has to be converted into a computer understandable form.

14.2.4.1 Coding *(*programming*)*

Coding is an important activity by which a programmer converts the systems specifications from the design stage into computer instructions referred to as programs. It is generally felt that the programs must be modular in nature. This helps in fast development, maintenance and future change if required.

14.2.4.2 Prototyping

Prototyping is the rapid development and testing of a working model of a product in an interactive and iterative process involving both systems analysts and end users. This working model or prototype, is a partially developed product that enables customers and developers to examine some aspects of the proposed product and decide if it is suitable for a finished product. Various types of prototyping exist.

a. Throw-away Prototyping

In throw-away prototyping, the prototype is discarded once the actual requirements have been understood and the final system is developed with a much clear understanding of user requirements.

b. Evolutionary Prototyping

In evolutionary prototyping, a functional prototype with minimal functionality is built in the beginning and is refined over time, as requirements are better understood.

c. Incremental Prototyping

In incremental prototyping, functional prototypes of the various subsystems are built and then integrated to form a complete system. In other words, the product is built as separate prototypes which are later merged into a final product.

14.2.4.3 Testing

Testing is the process of executing a program with the intent of finding an error. During testing, trial runs are done to check for errors and whether or not the new system meets the users' needs. Once source code has been generated, the software must be tested to uncover and correct as many errors as possible before delivery.

There are three sets of data that can be used to test the system: normal data and abnormal data.

- Normal data is data which the system will accept.
- Abnormal (erroneous) data is invalid data which the system will reject.
- Extreme data are data values that are chosen at the absolute limits of the normal range. This is to ensure that all normal values will be accepted and processed correctly.

Using these test data, the following test runs can be carried out:

a. Unit Testing

The individual units or modules are tested separately with prepared test data so that any errors can be corrected.

b. Integration Testing

The complete system is tested after the individual units have been tested and put together. This tests that separately developed modules/units work together as planned without error.

c. System Testing

The integrated system is tested to evaluate the system's compliance with specified requirements.

d. Black-Box Testing

Black-box testing is a test that relies on the input/output behavior of the system, without any assumptions to what is happening within the system. It examines some fundamental aspects of a system with little regard for the internal logical structure of the system. Black-box tests are used to demonstrate that system functions are operational, that input is properly accepted and output is correctly produced, while at the same time searching for errors in each function.

e. White Box Testing

White-box testing, also called glass-box testing, is a test that relies on information about how the system has been designed and constructed. It requires knowledge of the internal structure or implementation of the system. White-box tests are conducted to ensure that internal operations are performed according to specifications and all internal components have been adequately exercised.

14.2.5 Documentation

The job of the programmer does not end with the code or software instructions. The organization or users need to know how to get the best out of the system. This is done through documentation. System documentation ensures continuity of the system.

There are two types of documentation; user documentation and technical documentation.

a. User Documentation

User documentation is a complete description of the system from the user's point of view detailing how to use or operate the system. It could be a paper-based user manual or help incorporated into the software that can be accessed when the software is installed. User documentation always covers the following:

- A guide that describes what the system is supposed to do in non-technical terms

- Instructions for installing and running the program
- Definition for hardware and Operating System requirements
- The format of the output data
- Explanation of common error messages and how to recover from them
- Description of how to make backups against accidental data loss

b. Technical Documentation

Technical documentation is a description from the designer's point of view. Technical documentation often contains:

- Detail functioning of the software showing algorithms, formulae, source codes etc.
- Description of data structures
- Test plans and testing procedures
- User interface and reports
- Location and version of the software

14.2.6 Implementation

Implementation is the conversion from the use of the present (old) system to the operation of the new system. It involves:

- ✓ Installation of new system
- ✓ Loading of data into new system
- ✓ Education and training of users of the system

There are different types of conversions:

a. Direct cutover

The old system is completely replaced by the new one. Its disadvantage is that, if the new system fails, there is no back-up system, so data can be lost.

b. Pilot run

The new system is installed in one part of the business or organization. This allows the new system to be fully developed and tested. Once the pilot system is running successfully, the new system is introduced to all of the business/organization.

- Its advantages are that, if something goes wrong with the new system, only a small part of the organization is affected, and the staff that were part of the pilot scheme can help train other staff.
- As a disadvantage, there is no back-up system for the office/department doing the pilot, if things go wrong.

c. Parallel run

The old and new systems are operated alongside each other (in parallel) until new system is proven capable.

- It is advantageous in that, if the new system fails, the old system will act as a back-up. Also, the outputs from the old and new systems can be compared to check that the new system is running correctly.

- Its disadvantage is that, entering data into two systems, and running two systems together, takes a lot of extra time and effort.

d. Phased implementation

The new system is installed in phases (stages or steps) gradually replacing parts of the old system until eventually, the new system takes over.

- Its advantages are that, it allows users to gradually get used to the new system, and training of staff can be done in stages.
- Its disadvantage is that, if a part of the new system fails, there is no back-up system, so data can be lost

14.2.7 Maintenance

Maintenance is the general process of changing a system after delivery to correct faults, improve performance or adapt the system to a changing environment or business requirements. Maintenance is necessary to eliminate errors in the system during its working life and to tune the system to any variations in its working environment.
Maintenance can be adaptive, preventive, corrective or perfective.

a. Adaptive Maintenance

Adaptive maintenance focuses on adjusting a software product to properly interface with a changing environment. Changes are made to the system to provide a closer fit between the system and its environment.

b. Preventive Maintenance

Preventive maintenance aims in retaining the system's capabilities before the occurrence of any problem (e.g. system failure). It locates weaknesses in the system and provides repairs in order to avoid any eventual breakdown of the system.

c. Corrective Maintenance

Corrective maintenance aims in restoring a defective system to a required state. This implies that repairs are made after a breakdown of the system.

d. Perfective Maintenance

Perfective maintenance refers to enhancements to the product in order to either add new capabilities or modify existing functions.

14.3 SDLC Models

There are various SDLC models defined and designed which are followed during software development processes. These models are also referred to as "Software Development Process Models". Each model follows a series of steps unique to its type, in order to ensure success in the process of software development.

14.3.1 Waterfall Model

The waterfall model is the most common and classic of SDLC models. It illustrates the software development process in a linear sequential flow; hence it is also referred to as a linear-sequential

life cycle model. In the waterfall model, each phase must be completed in its entirety before the next phase can begin. At the end of each phase, a review takes place to determine if the project is on the right path and whether or not to continue or discard the project.

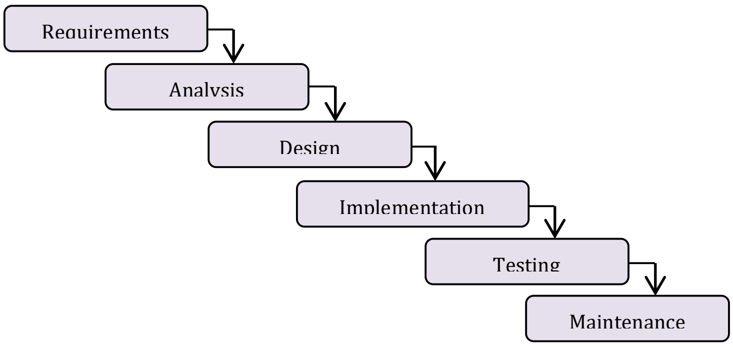


Figure 14.2: *Waterfall Model*

- ✓ ***Advantages***
  - Simple and easy to use.
  - Easy to manage due to the rigidity of the model – each phase has specific deliverables and a review process.
  - Phases are processed and completed one at a time.
  - Works well for smaller projects where requirements are very well understood.

- ✓ ***Disadvantages***
  - Adjusting scope during the life cycle can kill a project
  - No working software is produced until late during the life cycle.
  - High amounts of risk and uncertainty.
  - Poor model for complex and object-oriented projects.

14.3.2 V-Shape Model

The V-shape model is an SDLC model where execution of processes happens in a sequential manner in V-shape. It is also known as Verification and Validation model.

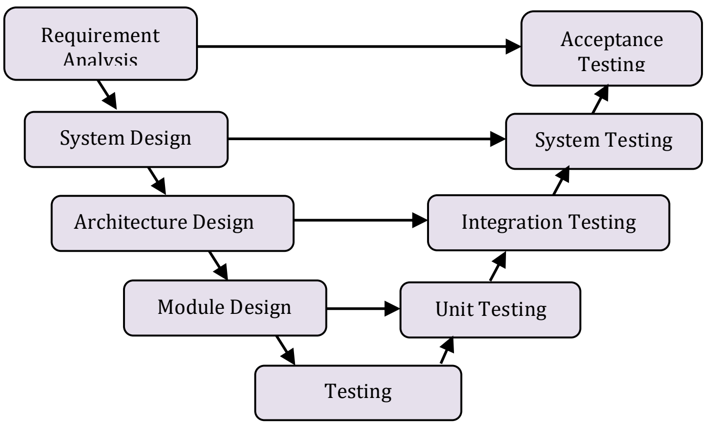


Figure 14.3: *V-Shape Model*

V-shape model is an extension of the waterfall model and is based on association of a testing phase for each corresponding development stage. This means that for every single phase in the

development cycle there is a directly associated testing phase. This is a highly disciplined model and next phase starts only after completion of the previous phase.

- ***Advantages***
  - Simple and easy to use.
  - Each phase has specific deliverables.
  - Higher chance of success over the waterfall model due to the development of test plans early on during the life cycle.
  - Works well for small projects where requirements are easily understood.

- ***Disadvantages***
  - Very rigid, like the waterfall model.
  - Little flexibility and adjusting scope is difficult and expensive.
  - Software is developed during the implementation phase, so no early prototypes of the software are produced.
  - Model doesn't provide a clear path for problems found during testing phases.

14.3.3 The Spiral Model

The spiral model has four phases: Planning, Risk Analysis, Engineering and Evaluation. A software project repeatedly passes through these phases in iterations (called Spirals in this model). In the baseline spiral, starting in the planning phase, requirements are gathered and risk is assessed. Each subsequent spiral builds on the baseline spiral.
Requirements are gathered during the planning phase. In the risk analysis phase, a process is undertaken to identify risk and alternate solutions. A prototype is produced at the end of the risk analysis phase.

- ***Advantages***
  - High amount of risk analysis
  - Good for large and mission-critical projects.
  - Software is produced early in the software life cycle.

- ***Disadvantages***
  - Can be a costly model to use.
  - Risk analysis requires highly specific expertise.
  - Project's success is highly dependent on the risk analysis phase.
  - Doesn't work well for smaller projects.

14.4 Software Design

Software design is the process of defining the architecture, components, modules, interfaces, and data for a software to satisfy specified requirements. It is the activity within the software development life cycle, where software requirements are analyzed in order to produce a description of the internal structure and organization of the system that will serve as the basis for its construction.
A software design must describe the architecture of the system - how the system is decomposed and organized into components and must describe the interfaces between these components. It

must also describe these components into a level of detail suitable for allowing their construction.

14.4.1 Design Strategies and Methods

Various general strategies can be used to help guide the design process. Common examples are top-down design and stepwise refinement.

14.4.1.1 Top-down Design

Looking at some problems as a whole can make them seem impossible to solve because they are so complex. For example, writing a word processor program or an operating system. Such complex problems can be solved by breaking them into smaller parts which are solved individually and the smaller solutions are assembled into a big solution. This process is called decomposition, divide and conquer, or more commonly top-down design.

***Definition**: Top-down design is the process of designing a solution to a problem by systematically breaking the problem into smaller, more manageable parts.*

The problem is broken down into smaller problems called sub-problems, which in turn are broken into smaller sub-problems, continuing until each sub-problem can be solved in a few steps.

Top-down design leads to modular development in which software modules are developed individually then the modules are combinedto form a solution to an overall problem.

***Definition**: A module is a self-contained entity that results when a problem is divided into sub-problems; each module corresponds to a sub-problem.*

Modular development of computer software:

a) *Makes large projects more manageable.*
   Smaller and less complex tasks are easier to understand than larger ones and are less demanding of resources.

b) *Is faster for large projects*.
   Different people work on different modules at the same time which speeds up the overall project.

c) *Leads to a higher quality product*
   Programmers with knowledge and skills in a specific area can be assigned to the parts of the project that require those skills.

d) *Makes it easier to find and correct errors*
   It is easy to isolate the part of the software that is causing trouble, making it easy to fix it.

e) *Increases the reusability of solutions*

Solutions to smaller problems are more likely to be useful elsewhere than solutions to bigger problems.

14.4.1.2 Stepwise Refinement

Stepwise refinement is the process of breaking a problem down through successive steps into smaller problems. It is an iterative process where each problem is decomposed and refined step by step.In stepwise refinement, the solution to a problem is first described in terms of high level functions, then, each function is broken down into details that are refined in successive steps until the whole problem is fully defined.

14.4.2 Design Notations

A large number of notations and languages exist to represent software design artefacts. Some are used mainly to describe the structural organization of a design, whereas others are used to represent the behaviour of such software systems. Notations for describing the structural organization of the system are static. For example, entity relationship diagrams, structure charts, class and object diagrams. Notations for describing the behavioural organization of the system are dynamic. Examples are data flow diagrams, flow charts, sequence diagrams, state transition diagrams and pseudo-code.

14.4.2.1 Data Flow Diagrams

A data-flow diagram (DFD) is a graphical representation of the "flow" of data through a system. A DFD shows the flow of data among a set of processes.

14.4.2.1.1 Flowcharts

They are used to show the flow of control and the associated actions to be performed.

14.4.2.1.2 State Transition Diagrams

They are used to show the flow of control from state to state in a state machine.

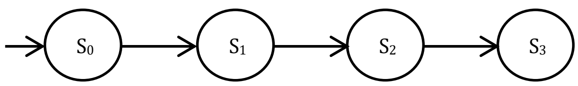


Figure 14.4: *State transition diagrams*

14.5 Software Reuse

Software reuse is the process of implementing or updating software using existing software assets. Reusable software assets include more than just codes. Requirements, designs, models, algorithms, tests, documents, and many other products of the software process can be reused.
Software reuse makes programming easier because you only need to develop the solution to a problem once; then you can call up that code whenever you need it. Modules developed as part of one project, can be reused later as parts of other projects, modified if necessary to fit new situations. Most computer systems are filled with layers of short programming modules that are constantly reused in different situations.
Software reuse can be horizontal or vertical.

14.5.1 Horizontal Reuse

Horizontal reuse refers to software components used across a wide variety of applications. This type of reuse includes library of components, string manipulation routines, or graphical user interface functions.

14.5.2 Vertical Reuse

Vertical reuse, in another way, deals with the reuse of system functional areas, or domains that can be shared between systems with similar functionality.

Advantages of Software Reuse

The main advantages of software reuse include:

- ✓ Increase in software productivity
- ✓ Reduced software development time
- ✓ Reduced software development and maintenance costs
- ✓ Improved software system interoperability
- ✓ Software is developed with fewer people
- ✓ Easy movement of personnel from project to project,
- ✓ Production of more standardized software,
- ✓ Production of better quality software

14.6 Project Management

A project is an endeavour to accomplish a specific objective through a unique set of interrelated tasks and the effective utilization of resources. A project has a definite start and finish time, well-defined outcomes or performance goals, and consumes scarce resources such as money, personnel, material, and equipment.
A project is considered constrained by three functions:

- ✓ *Scope*: what it is intended to accomplish. In other words, the customer's requirements for the project.
- ✓ *Time allocation*: the time schedule for the project.
- ✓ *Cost*: the money, budget, and resources for a project.

These three functions are called the Triple Constraint. The relationship between them is represented using the Project Management Triangle.

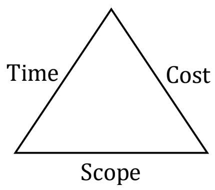


Figure 14.5: *Project management triangle*

The Project Management Triangle visualizes the fact that time, cost and scope of a project are interdependent; changing one of them, causes changes in the other two. For example, if you want to shorten a schedule, you can hire more resources which would increase cost, or reduce customer requirements which would affect quality. This simply means "*you can have any two of quick, good or cheap, but not all three*."

Projects are implemented through project management. Project management is defined as the application of knowledge, skills, tools and techniques to activities of a project for the achievement of the project's objectives/requirements. Project management ensures that an acceptable system is developed within time and budget.

14.7 Project Management Terms

a) ***Task/Activity***: anything that needs to be done that requires time and consumes resources.
b) ***Dependent task***: a task that can only begin after a previous one is finished. For example, roofing a house depends on the construction of the walls.
c) ***Slack time or float time***: the amount of delay that can be tolerated between the starting time and completion time of a task without causing a delay in the completion date of the entire project. If we have tasks A and B that start at the same time and task C that is dependent on both tasks A and B. If task A takes 3 days and task B takes 5 days, then task A has 2 days slack time. That is, it can run for 2 days before it affects the planned starting time for task C.
d) ***Lag time***: the delay or amount of time that passes between the end of one activity and the beginning of another if the two are dependent. For example, if task A is laying of cement blocks and dependent task B is building the walls of the house, there would be some lag time between the end of task A and the start of task B to let the blocks get dry.
e) ***Lead time***: occurs when a task should theoretically wait for its predecessor to finish, but can actually start a little early. The time that the tasks overlap is lead time.
f) ***Milestone***: an event that signifies the accomplishment or completion of a major deliverable during a project.
g) ***Deliverable***: some concrete thing which is to be delivered, to the client or internally to the development team.
h) ***Levelling***: the process of adjusting tasks to match resources available. There are two techniques for levelling: task delay and task split.
i) ***Critical path***: a sequence of dependent tasks that have the largest sum of most likely durations. The critical path determines the earliest possible completion date of the project.
j) ***Critical task***: a task found on the critical path. A critical task cannot be delayed without delaying the entire project schedule.
k) ***Work Breakdown Structure***: a hierarchical decomposition of the project into phases, activities, and tasks.
l) ***Project management plan*** (PMP): a document that describes how the project is to be executed, monitored and controlled, which includes creating a project work breakdown structure, identifying and planning to mitigate risk, identifying manners in which to effectively communicate with stakeholders and other project team members, and developing a plan to manage changes.

14.8 Project Life Cycle

The activities related to a project can be structured and grouped into stages according to the aim of the activities. A typical project goes through the following stages called project life cycle: initiation, planning, execution, monitoring and control, and closing.

14.8.1 Initiation

Project initiation determines the main objective of the project and forms a clear understanding about the necessity and suitability of the project. This stage answers the questions "*what?*" and "w*hy?*" Common activities at this stage are:

- ✓ Identification and initial analysis of the business needs.
- ✓ Determination of the main objective(s).
- ✓ Resource analysis (people, equipment, financial; needs and availability).
- ✓ Composition of the project charter.

(*Project charter - document issued by the project initiator or sponsor that formally authorizes the existence of a project, and provides the project manager with the authority to apply organizational resources to project activities*.)

14.8.2 Planning

Project planning involves the project plan development and approval. It determines an optimal scheme/algorithm for project execution. This stage answers the question "*how?*" The main activities at this stage are:

- ✓ Needs analysis
- ✓ Description of the project (including determination of activities and necessary resources)
- ✓ Composition of project plan
- ✓ Planning and performing necessary PR-activities.

(*PR is the practice of managing the spread of information between an individual or an organization and the public. Public relations activities include: launchings, media conferences, sales promotions, open day, product testing, websites, press release, newsletters*.)

14.8.3 Execution

Project execution is the phase within which the deliverables are physically constructed and presented to the customer for acceptance. It integrates people and other resources to carry out the project management plan for the project. The activities undertaken to construct each deliverable will vary depending on the type of project being undertaken. Main activities are:

- ✓ Starting up the execution.
- ✓ Building the deliverables
- ✓ Day-to-day management and reporting

14.8.4 Monitoring and Control

Project control measures and monitors progress to identify variances from the project management plan so that corrective action can be taken when necessary to meet project objectives. Control occurs throughout the duration of the project and has a range relatively similar to that of execution. While the project is being executed, a series of management processes are undertaken to monitor and control the deliverables being output by the project. This includes:

- ✓ Requesting, evaluating and approving changes to the project scope, deliverables, timescales or resources (change management)
- ✓ Controlling the amount of time spent undertaking each activity within the project (time management)
- ✓ Identifying, approving and paying cost/expenses incurred on project (cost management)

- ✓ Reviewing deliverable quality (quality management)
- ✓ Identifying, quantifying and managing risks to the project (risk management)
- ✓ Identifying and handling issues currently affecting the ability of the project to produce the required deliverables (issue management)
- ✓ Measuring each deliverable produced against acceptance criteria (acceptance management)
- ✓ Handling sourcing of products from an external supplier (procurement management)
- ✓ Identifying, creating, and reviewing communication messages within the project (communication management)
- ✓ Performing a phase review at the end of execution to ensure the project has achieved its objectives as planned.

14.8.5 Closing

Once all the deliverables have been produced and the customer has accepted the final solution, the project is ready for closure.

Project closure involves:

- ✓ releasing the final deliverables to the customer
- ✓ handing over project documentation to the business
- ✓ terminating supplier contracts,
- ✓ releasing project resources
- ✓ communicating the closure of the project to all stakeholders

14.9 Project Analysis and Scheduling

Project scheduling is the process of converting a general or outline plan for a project into a time-based schedule based on the available resources and time constraints. Different techniques exist for analysing and scheduling project activities.

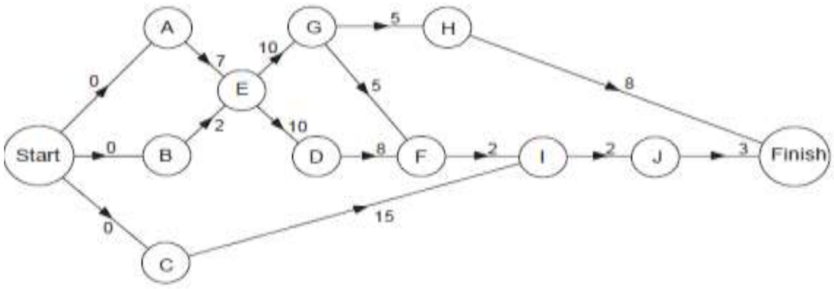
14.9.1 Critical Path Method (CPM)

CPM is an analysis technique used to predict project duration by analyzing which sequence of activities (which path) has the least amount of scheduling flexibility (the least amount of total float). Early dates are calculated by means of a forward pass using a specified start date while late dates are calculated by means of a backward pass starting from a specified completion date, usually the forward pass's calculated project early finish date.

- ***Forward pass**:* The calculation of the early start and early finish dates for the uncompleted portions of all network activities, determined by working forward through the schedule network logic from the project's start date.
- ***Backward pass**:* The calculation of late finish and late start dates for the uncompleted portions of all schedule activities, determined by working backward through the schedule network logic from the project's end date.

CPM models the events and activities of a project as a network. Activities are depicted as nodes on the network and events that signify the beginning or ending of activities are depicted as arcs or lines between the nodes.

a)



b)

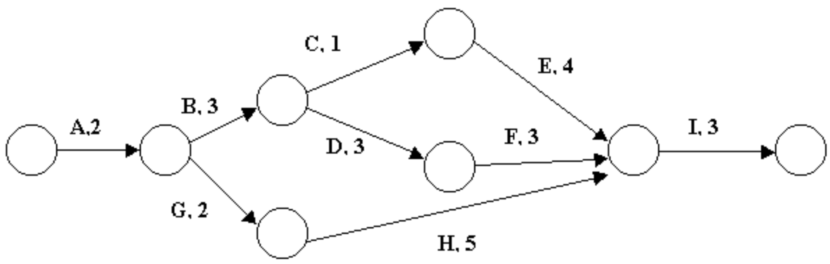


Figure 14.6: *Network diagrams*

In the above network diagram,

- ✓ Tasks are lettered from A to J. There are two ways of representing the activities on a network diagram: (a) activity on node and (b) activity on arc.
- ✓ Joining task A to E shows that task A must be completed before task E can be started.
- ✓ Joining tasks A and B to E shows that both task A and B must be completed before task E can be started.
- ✓ The number marked on each arc (arrow) shows the duration of the task from which the arc starts.

The critical path can be identified by determining the following four parameters for each activity:

- ✓ ***ES – earliest start time***: the earliest time at which an activity can begin given that its predecessor activities must be completed first.
- ✓ ***EF – earliest finish time***, equal to the earliest start time for the activity plus the time required to complete the activity.
- ✓ ***LF – latest finish time***: the latest time at which an activity can be completed without delaying the project.
- ✓ ***LS – Latest start time***, equal to the latest finish time minus the time required to complete the activity.

Activities with the same earliest and latest start times (ES=LS) or with same earliest and latest finish times (EF=LF) define the critical path. This means that these activities have a float time of 0.

For the above network diagram, we have:

Activity	**Duration**	**Start times**	

		Earliest	**Latest**	**Float time**
A	7	0	0	0
B	2	0	5	5
C	15	0	12	12
E	10	7	7	0
D	8	17	17	0
F	2	25	25	0
G	5	17	19	2
H	8	22	24	2
I	2	27	27	0
J	3	29	29	0
Finish		32	32	

The critical path is A-E-D-F-I-J

The total estimated duration of the project = sum of duration of critical tasks = $7 + 10 + 8 + 2 + 2 + 3 = 32\ days$

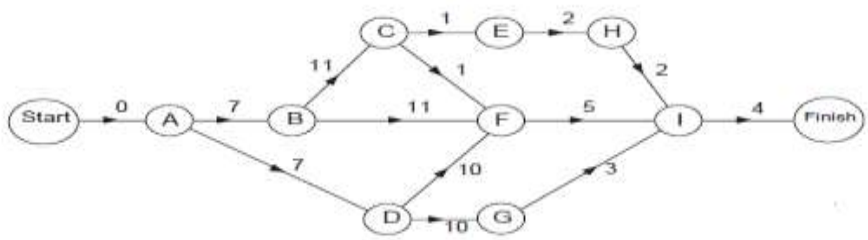
Exercise: Given the task description table below:

Activity	Duration	Precedence
A	3	-
B	3	A
C	4	-
D	1	C
E	3	B, D
F	2	A, B, D
G	2	C, F
H	4	G
I	1	C
J	3	E, G
K	5	F, H, I

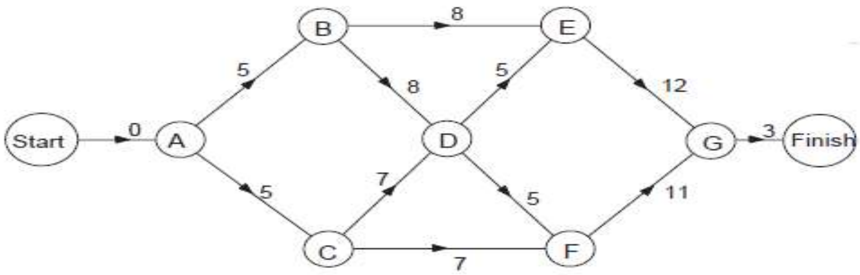
1. Draw the corresponding PERT diagram for this project
2. Determine the critical path
3. Calculate the total estimated duration of the project
4. State the float time for all non-critical activities

Exercise 2: Find the critical path for each of the activity networks below.

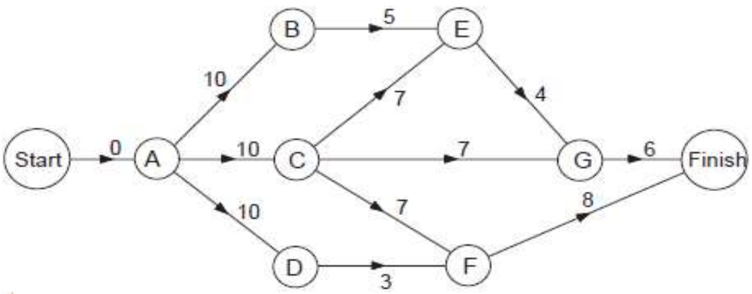
a.



b.



c.



$\boxed{\textit{Remark!}}$ CPM was developed for complex but fairly routine projects with minimal uncertainty in project completion times. For less routine projects there is more uncertainty in the completion times, and this uncertainty limits the usefulness of the deterministic CPM model. An alternative to CPM is the PERT (Program Evaluation and Review Technique) project planning model.

14.9.2 Program Evaluation and Review Technique (PERT)

PERT is an event-oriented network analysis technique used to estimate project duration when there is a high degree of uncertainty with the individual activity duration estimates. Each activity is assigned three time estimates

m= most likely time estimate, (mode)

a = optimistic time estimate (best case)

b = pessimistic time estimate (worst case)

These three estimates are then used to calculate a weighted duration for each task by using the formula$T_E = (a + 4m + b)/6$

The weighted durations are then used as a more realistic estimate of task durations for constructing a PERT chart (network diagram).

Using PERT, the probability of completing the project by a certain date t, can now be found by finding

$$P(t \geq T) = \frac{t - T}{\sqrt{\sigma_T{}^2}} = \frac{t - T}{\sigma}$$

Where

$T$is the expected completion time of the project

And

$$\sigma_T{}^2 = \left(\frac{b - a}{6}\right)^2$$

is the variance of T = Σ(variances of activities on the critical path).

Example:

If a project's expected completion time is $T = 246\ days$ with its variance$\sigma_T{}^2 = 25$, then what is the probability that the project

a) is actually completed within 246 days?
b) is actually completed within 240 days?
c) is actually completed within 256 days

Solution

a) $t = 246, T = 246$ and $\sigma_T{}^2 = 25$

$$P(t \geq T) = \frac{246 - 246}{25} = 0$$

b) $t = 240,\ T = 246$ and $\sigma_T{}^2 = 25$

$$P(t < T) = \frac{240 - 246}{25} = \frac{-6}{25} = -0.24$$

$$\therefore P(t \leq T) = 1 - (P(t < 0.24)) = 1 - 0.24 = 0.76$$

c) $t = 256,\ T = 246$ and $\sigma_T{}^2 = 25$

$$P(t \geq T) = \frac{256 - 246}{25} = \frac{6}{25} = 0.24$$

Definition: *A PERT chart/diagram is a graphic illustration of a project as a network diagram consisting of numbered nodes (either circles or rectangles) representing events, or milestones in the project linked by labeledvectors (directional lines) representing tasks in the project. The direction of the arrows on the lines indicates the sequence of tasks.*

14.9.3 Comparison between CPM and PERT

CPM task durations are known with certainty. CPM is therefore said to be deterministic while PERT is probabilistic.

	CPM	**PERT**
1	Uses network, calculate float or slack, identify critical path and activities, guides to	Same as CPM

	monitor and controlling project	
2	Uses one value of activity time	Requires 3 estimates of activity time Calculates standard deviation and variance of time
3	Used where times can be estimated with confidence, familiar activities	Used where times cannot be estimated with confidence. Unfamiliar or new activities
4	Minimizing cost is more important	Meeting time target or estimating percent completion is more important

Table 14.1: *Comparison between CPM and PERT chart*

14.10 Gantt Chart

A Gantt chart is a horizontal bar graph that helps plan and monitor project development or resource allocation on a horizontal time scale. It depicts project tasks against a calendar.

A Gantt chart is constructed with a horizontal axis representing the total time span of the project, broken down into increments (days, weeks, or months) and a vertical axis representing the tasks that make up the project. Horizontal bars of varying lengths represent the sequences, timing, and time span for each task. The bar spans may overlap, as, for example, you may conduct research and choose software during the same time span. As the project progresses, secondary bars, arrowheads, or darkened bars may be added to indicate completed tasks, or the portions of tasks that have been completed. A vertical line is used to represent the report date.

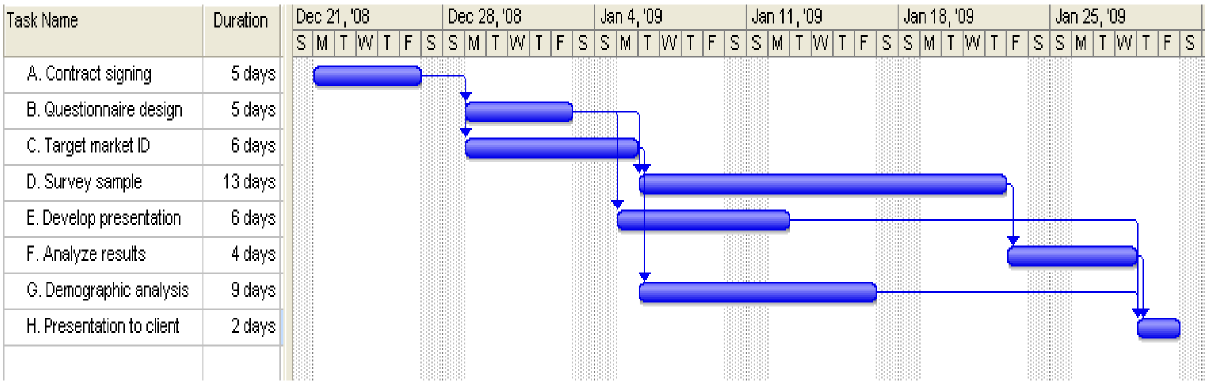


Figure 14.7: *Gantt chart*

The critical path is A-C-D-F-H = 5+6+4+2 = 17 days

Drill Questions:

1: A project has been defined to contain the following list of activities along with their required times for completion.

Activity No.	Activity	Time (weeks)	Immediate Predecessors

1	Collect requirements	3	
2	Analyze processes	2	1
3	Analyze data	2	2
4	Design processes	6	2
5	Design data	3	3
6	Design screens	2	3,4
7	Design reports	4	4,5
8	Program	5	6,7
9	Test and Document	7	7
10	Install	2	8,9

a. Draw a network diagram for this project.
b. Calculate the earliest completion time of the project
c. Show the critical path.
d. What would happen if activity 6 were revised to take 6 weeks instead of 2 weeks?
e. Construct a Gantt chart for this project

2: At 4:30 pm one day CRTV news team hears of a Government Minister resigning. They wish to prepare an item on the event for that evening's 6 o'clock news. The table below list the jobs needed to prepare this news item, the time each job takes and the constraints on when the job can commence.

Job	**Time needed**	**Constraints**
A - Interview the resigning minister	15 mins.	Starts at 4:30 pm
B - Film the ministry	20 mins.	None
C - Get reactions from regions	25 mins.	Cannot start until A and B are completed
D - Review possible replacements	40 mins	Cannot start until B is completed
E- Review the minister's career	25 mins	Cannot start until A is completed
F- Prepare film for archives	20 mins.	Cannot start until E and C are completed
G- Edit	20 mins.	Cannot start until A, B, C, D, E and F are completed

a. Construct an activity network for this problem and by finding the critical path in your network, show that the news item can be ready before 6 pm that day.
b. Construct the corresponding Gantt chart for this project.

3.(i)(a) write out the steps involved in SDLC in an annotated diagram.
(b) Distinguish between piecemeal and direct system implementation methods.
(c) Describe one other alternative method of building a system.
(ii)(a) State two functions of a project manager in project management.
(b) With respect to project management, describe a PERT chart, stating all the elements involved.
(c) What is the importance of identitfying the critical path of a project?
(iii) Explain the following project management terminologies:
(a) Project (b) Critical path (c) Gantt chart
(iv)(a) What is prototyping?
(b) Give two reasons for prototyping. (**CGCE 2015**)

4.(i)(a) What is SDLC?
(b) Give two advantages of direct system implementation method
(c) State three activities involved in the design stage of SDLC.
(ii)(a) What is the role of a system analyst in an organisation?
(b) Describe two ways by which a system analyst can get information about a system to be developed.
(iii) Work can be classified into operations or project. Although in some cases, they overlap.
(a) Give ant TWO characteristics common to operations and project.
(b) Make a difference between multiprogramming and project. (**CGCE 2016**)

5.(a) Define the term SDLC.
(b) List four stages of the SDLC.
(c) For any two of the stages given in (**b**) above, briefly describe the major tasks involved (**Q8i, CGCE 2014**)

PART 2: PRACTICALS

1 MICROSOFT ACCESS

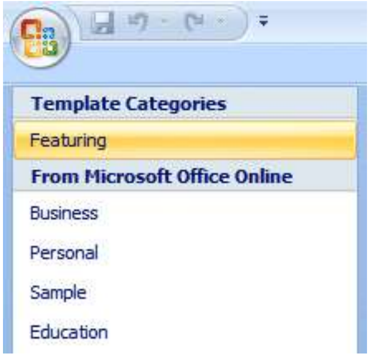
In this section we shall focus most on MS Access 2007 but the same procedures can be applied on MS 2010 and 2013 though you may have slight changes in the GUI. Higher versions of MS Access display more icons and simplifies work for users. But for accessibility reasons and convenience sake, we would guide you through using MS Access 2007. In case you have any other version of MS Access, just try apply the same instructions.

1.1 Starting Access 2007

Double-click on the **Access2007** icon on the **Windows** desktop (**seeright**), or **click**-on the **Start** button in the **lower left corner** of the screen, then **click**-on **Programs,** and then **click**-on **Microsoft Access 2007**.

The **Getting Started with Microsoft Office Access** screen will **appear (***image below*).

For previous Access users: The above menu screen is **new** in **Access2007**.Take a few minutes to peruse this screen. You will notice that (on the top left of the screen) that the "old" Access Templates (already created databases) are still available.
As we move through this tutorial, many features of "old" Access will be familiar to you.



Left Mouse Button

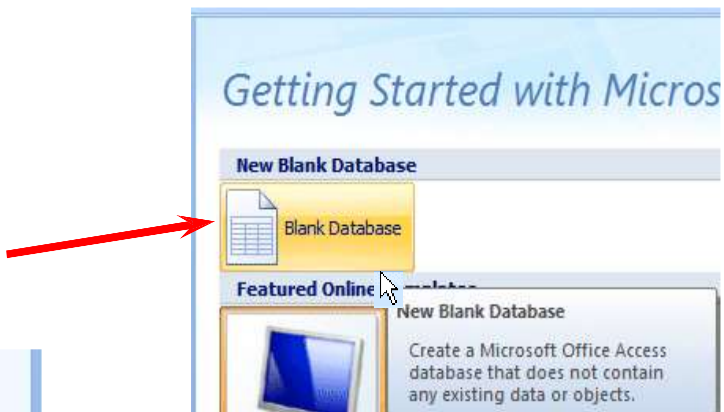
N.B: *In this tutorial, whenever we indicate that you need to click the mouse, it will mean to click the left mouse button–unless we indicate that you should click the RIGHT mouse button. So, always "click left" unless we tell you otherwise.*

1.1.1 Creating anAccess2007 Database

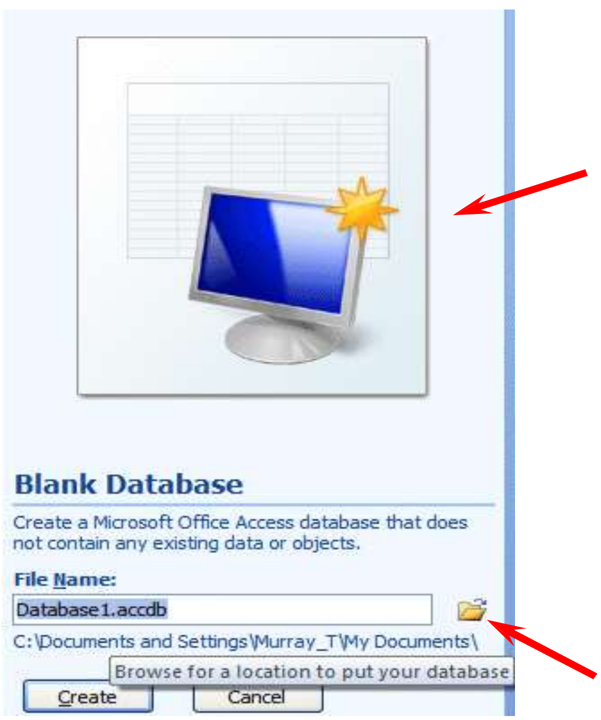
This tutorial will assist you in creating a database that includes the features most often used in databases. Once you gain skill with the database you create, you will be able to use and understand the already created Microsoft Access databases mentioned on the last page of this chapter.

We'll begin with a Blank Database and increase our database knowledge with each step.

Look at the **center** of your Access screen. You will see – **Getting Started with Microsoft Office Access**. Below the title, you will see a Blank Database button.



Click the **Blank Database button**.



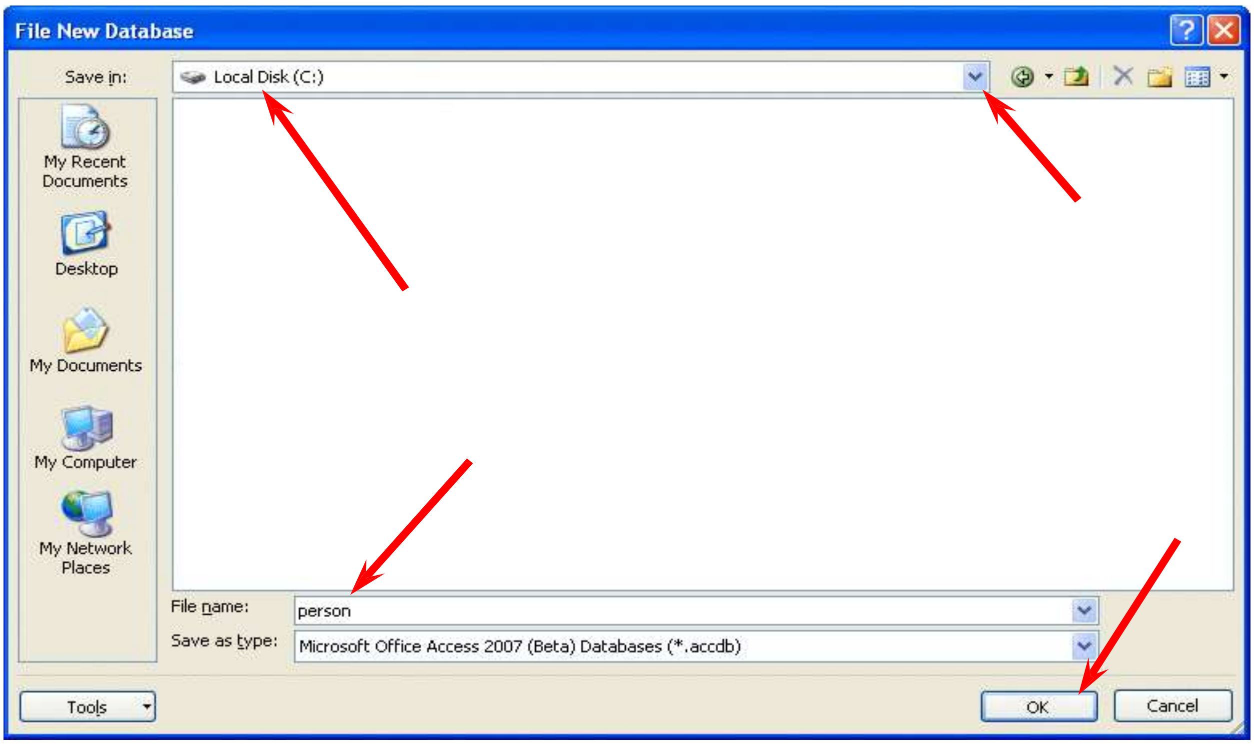
As soon as you **click** the **Blank Database button**, the right side of your Access screen will change and look like the image on the left.

1.1.2 Saving your work

One peculiar aspect about Access databases is that it requires you to save your database as soon as you enter the program. You can **save** your work on a *floppy diskette* in *Drive A***,** on a *USB key/Flash Drive* or on your *Drive C: Hard Disk*, or in some other drive.

Please save to one of these areas and substitute your Drive in the instructions.

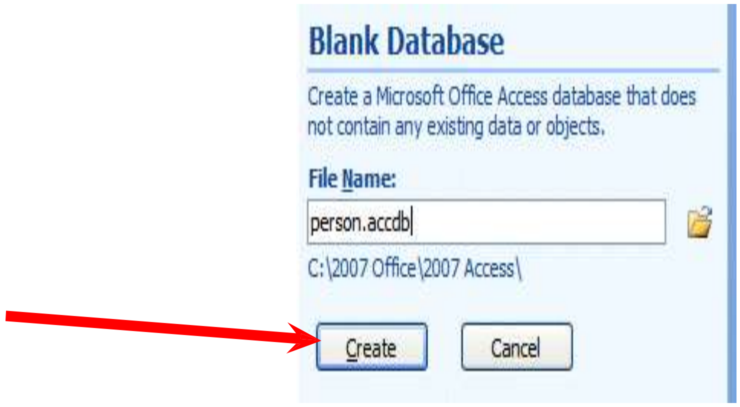
To choose the Drive, on which you will save your Access database, **click** the **small folder** to the **right** of **File Name (***as shown in the second figure above***):**
A **New File Database** menu screen similar to the one below will appear when you click the folder.



In the **upper left corner** of the **File New Database** menu screen that appears, you will see a **Save in** area (*see top left arrow above*). **Click** on the **small down arrow** on the **right** and it will show you the various disk drives available on which you can save (*see right upper arrow above*). Point to the drive on which you want to save your database, and click on it. If you choose the *3½ Floppy (A:)*, make sure you have a *formatted disk* in the A drive. Else if you choose the *C: drive*, choose the folder in which you want to save by **double clicking** on the folder. Your selection should now **appear** in the **Save in** area.
Next click in the area to the right of **File Name**. **Delete** any text that is entered in the area and then type-in the word person or any other user defined file name (*as shown at the bottom of the above image, see lower left arrow*).

Now click-on the **OK button** or tap the **Enter** key (*see lower right arrow on last page*).

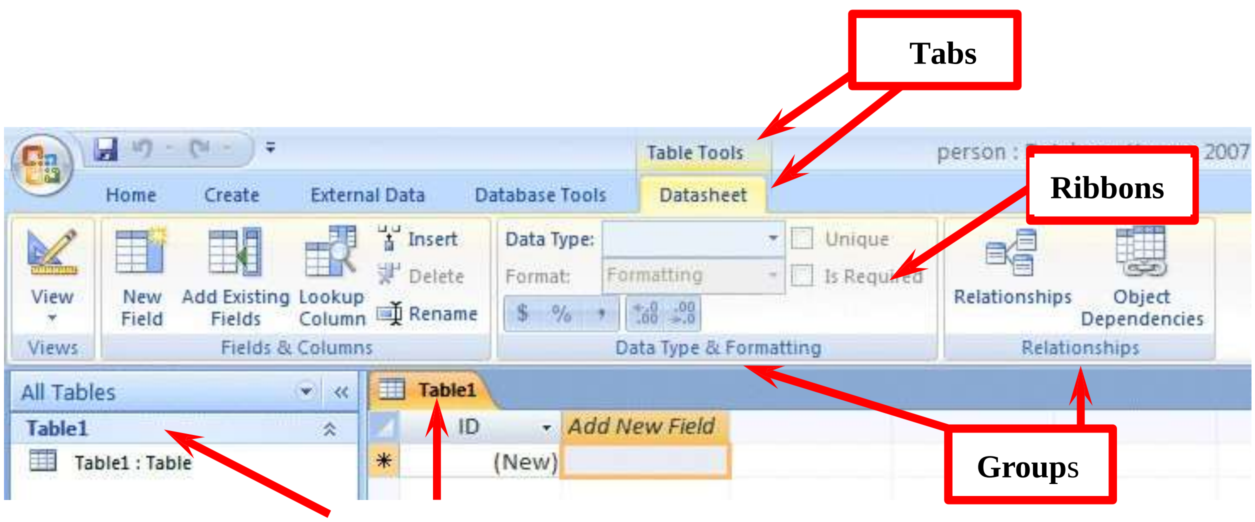
You will now return to the **Getting Started with Microsoft Office Access screen**. On the **right side** of the screen, you will see your data base **file name** and **below** it, the drive on which you will create your database.



Click the **Create button**.

1.1.3 Creating a Table

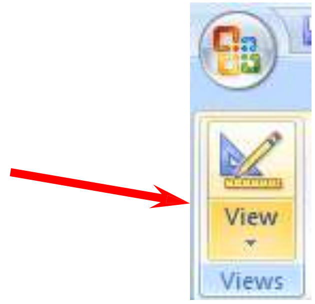
When you click the **Create button,** your Access2007 screen will change to the image below. This is the "*new look*" in **2007 Office**. You will now see **Tabs** and **Ribbons** that automatically appear for the area in Access on which you're working. Instead of a Menu Bar and drop down selections, you'll now see these new features.



When we clicked the create button, access assumed we desired to create with in our person database, another database which is called **table1**. You'll notice that at the top of the above image that the table tools and datasheet tabs appeared to assist you. The ribbon below these tabs is composed of groups of selections you'll use as you create your table. We'll be working with these tabs/ribbons throughout this tutorial.

In the lower portion of the above image you'll see selections that indicate we are creating a new Table.

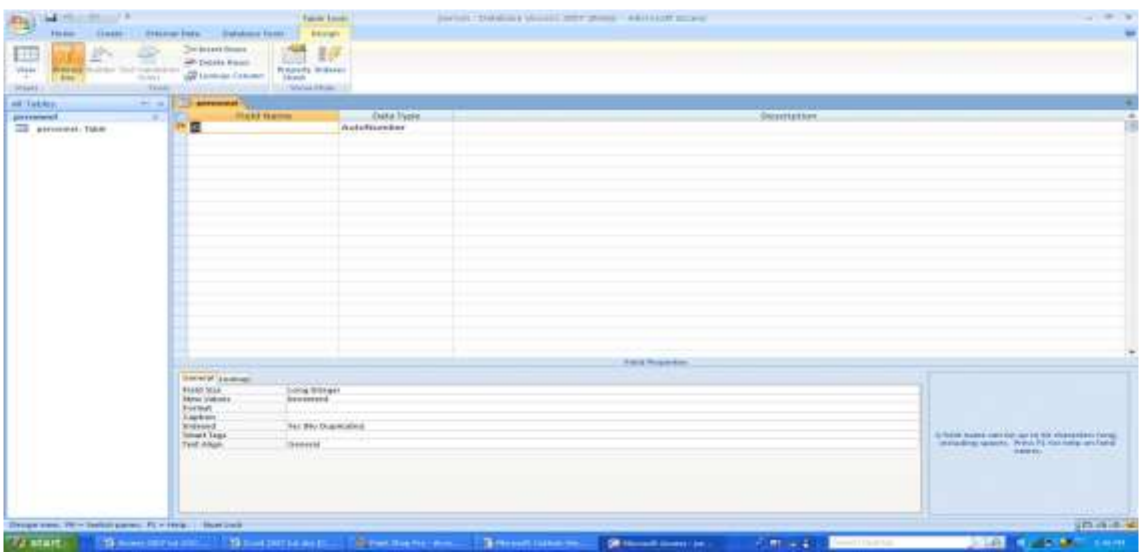
On the left of the Table Tools-Datasheet Tab/Ribbon you'll see a View button. Click the **View button**.



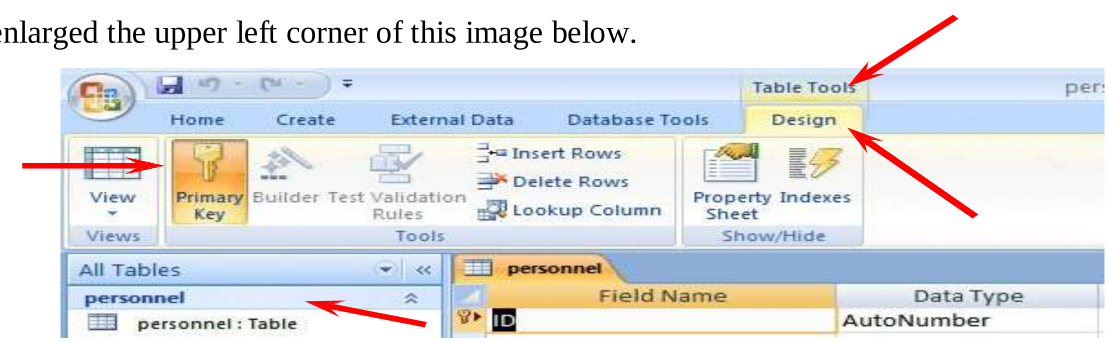
When you click the View button, the image on the left will appear. Since we want to create or design a new Table, we'll click the Design View selection.

A Save As menu screen will appear similar to the image on the right. Type personnel in the "Table Name:" area and then click the **OK** button.

Your Access 2007 screen will now change gain – to the image below.



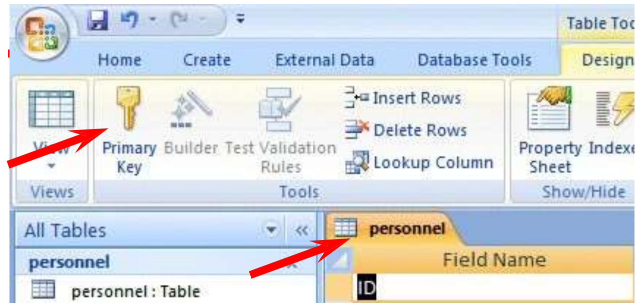
We've enlarged the upper left corner of this image below.



N.B: *In the image above, our **Table Tools Tab** still appears. However, because we are now in the Design process, the **lower Tab/Ribbon** has changed to **Design** – to assist us with designing our Table.*

In the image on the last page you will notice that the **Primary Key** button is "*orange*" and in our Personnel Table, that it is also "*orange*" – with a little key to the left of ID.

In database language this is called "*Keying*."

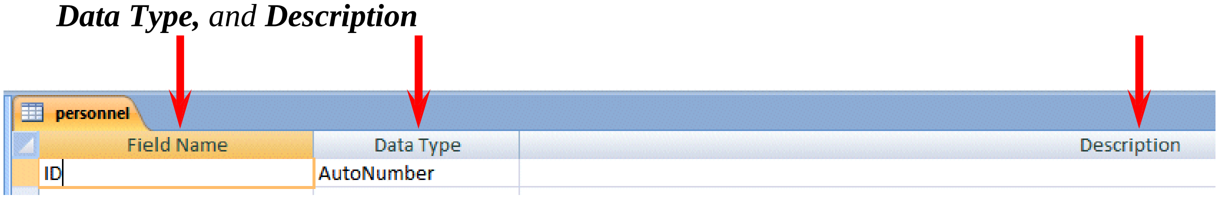


Keying, or indexing, in some aspects is advanced. You can get a good description by searching in "*Help*" for the usage of Keying or Primary Key.

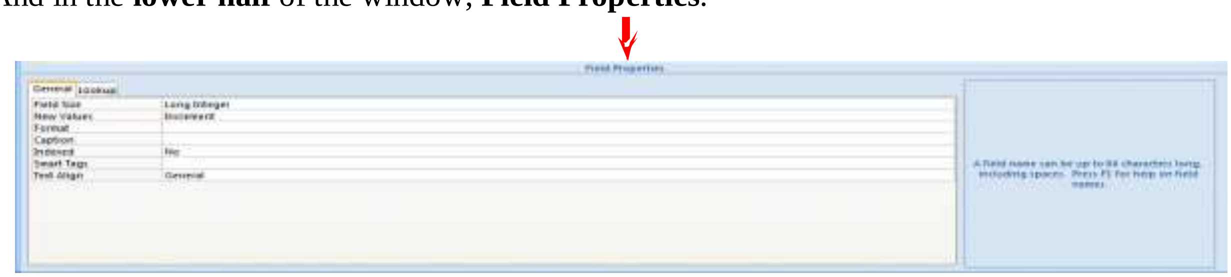
To "*turn-off*" the Primary Key, click the Primary Key button. You'll notice that the Primary Key button won't be "*orange*" and the little key gone from the left of ID.

If the **Personnel Table** image does not fill the screen, click-on The *small square* between the "*minus and the X*" in the *top right edge* of the screen (*see arrow position on image on right*). This will *maximize* the dimension of the screen.

*Under the Blue Bar at the top of the Design screen, there are (3) things: **Field Name, Data Type,** and **Description***



And in the **lower half** of the window; **Field Properties**.

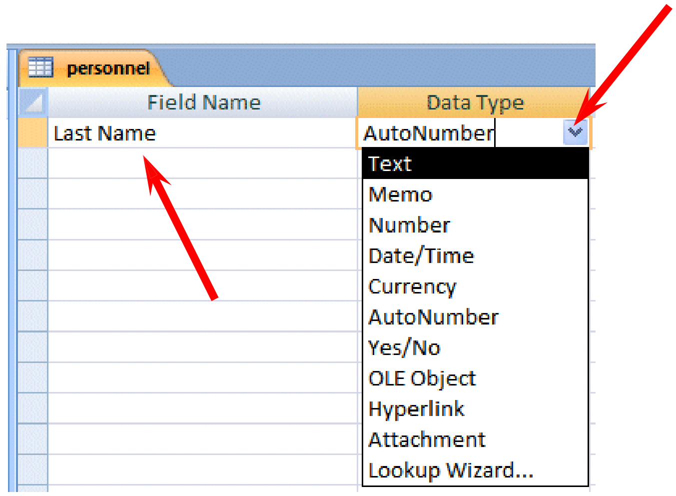


We'll be **creating** the **Field Names** that make up a **database**. This is similar to creating a **blank personnel form (onpaper)** that will be "**filled-in**" for each employee (Name, Address, Phone Number, etc.).The **areas that will be filled in** are called **Fields** in a database.
When you fill in all of the fields for a person, the individual "**forms**" are called **records** in a database. There will be a record, or form, for each employee. All the forms, together, make up a Table(database).

So let's create a personnel database.

N.B: When **creating** a **database** it is always best to "*break down*" a **field** in to its "*smallest parts*." For example – **Name** would break down into **First Name**, and **Last Name** (*you could also have **Middle Name**, **Title**, etc.*) Address would break down into **Street Address**, **City**, **State**, and **Zip** (*you could also have **Apartment Number** etc.*) Because we are working in Access 2007 it will be very simple to put the fields back together with a few mouse clicks when the need arises.
Trust us!!! This will save you a lot of time later on.

Look at the image on the right. Click - in the area or space under **Field Name** and type-in **Last Name** (*to replace ID*). Tap Enter or click – in the area to the right under **Data Type**. The cursor now moves to the right under **Data Type**. Click the down arrow under **Data Type** on right (*see image on right*). Your design screen should look like the one on the right.



*Now we'll talk about **Data Types**.*

1.2 DataType

Text You may type in any alphabetical/numerical data that you desire-up to a maximum of 255 characters. As indicated, this is a text field, so you can't do mathematical calculations. Examples of Text data are: *names, addresses, stock numbers, room numbers, zip codes etc*.

Memo This field is for lots of text. You can have up to 32,000 characters.

Number This field is for numbers where you want to add, subtract, multiply, divide, perform averages and do numerical calculations. This field can be a very large size, so when we get to **Field Properties**, we'll talk about "*sizing*" this field so it doesn't take up too much "space" in storage.

Date/Time Dates and Times. You may format these later as you may desire.

Currency Dollars ($). You may format these later as you may desire.

AutoNumber This field is an "*automatic*" counter that assigns a number each time you put data into a new field.

Yes/No This is a "True/False" or "Yes/No" type of field.

OLE Object this means *Object Link Embedding* which indicates you can insert a graphic, picture, sound, etc. in a desired record. Pretty neat to put a photograph in a personnel record or a picture of an inventory item in the stock record.

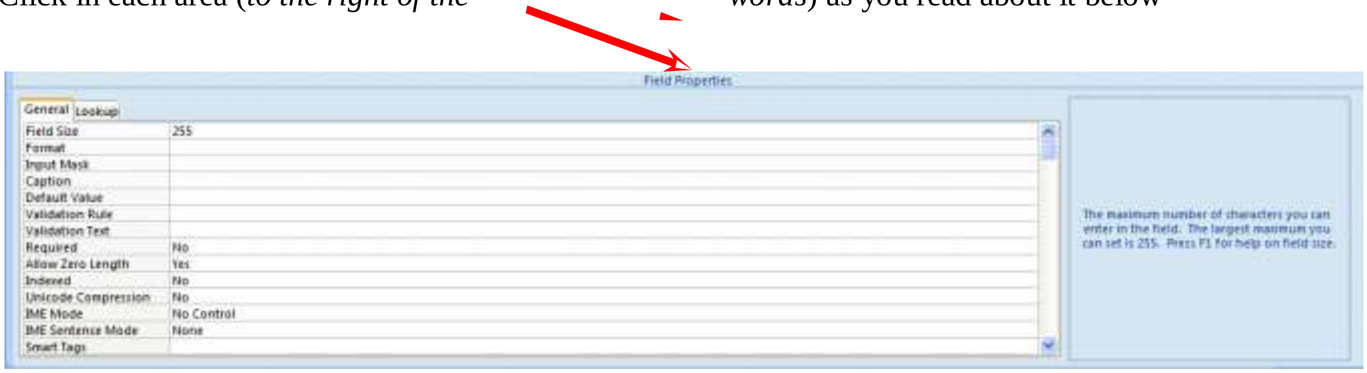
Since this is a simple, introductory Access tutorial, we won't work with **Hyperlinks**, **Attachments**, or **Lookup Wizards**.

We'll leave **Last Name** as a **Text Data Type**. To the right under **Description Field** you may make any remarks you feel are appropriate to someone who may want to know how/why you designed the field as you did.

Now, you would have noticed in the lower part of the screen, under **Field Properties,** that a box appeared when you selected the **Text Data Type**. This box is tailored to the **Text Data Type** that you selected above. Your **Field Properties** should look like the one below when you finish executing the recommended **steps**

1.3 Field Properties

Click in each area (*to the right of the* *words*) as you read about it below



Field Size Is currently set to 255 characters. That's pretty large for a name. So, click in this area and change the number to 25 (*you can make this larger or smaller later if need be*).

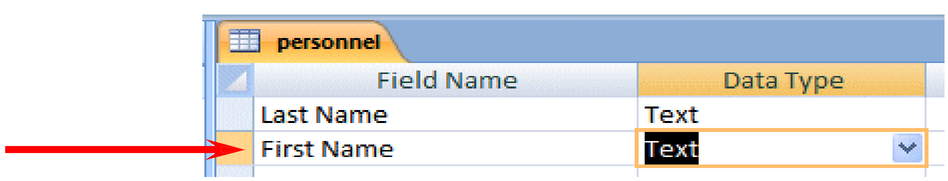
Format Now click in the **Format Area**. Next tap the F1 function key to activate *Help*. Since you are in the **Format Area**, Help will be tailored to this area. When the *Help* window appears, click **Format Property**.

Now **click Text and Memo Data Types**. Notice that there are several choices to make your character supper or lowercase. This gives you an idea of some formats.
We'll use one later!

Now click – on the "X" in the top right edge of the Microsoft Access Help – **Format Property** Window to close it.

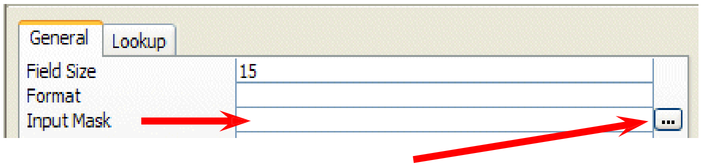
Input Mask We'll come back to this feature later.
Caption Look at the Light Blue Help area to the right. It explains about Caption.
Default Value We'll come back to this feature later.
Validation Rule We'll come back to this feature later.
Validation Text We'll come back to this feature later.
Required Look at the Light Blue Help area to the right.
Allow Zero Length Look at the Light Blue Help area to the right.
Indexed Look at the Light Blue Help area to the right.
Unicode Compression Look at the Light Blue Help area to the right.
IME Mode Look at the Light Blue Help area to the right.
IME Sentence Mode Look at the Light Blue Help area to the right.
Smart Tags Look at the Light Blue Help area to the right.

Now we'll repeat this process and create different **Field Names** and **Data Types**. **Type-in** the **Field Names** as indicated below and set them to the **Data Types** and **Sizes** indicated. Start each new Field Name and Data Type below the previous field (*see example below*)



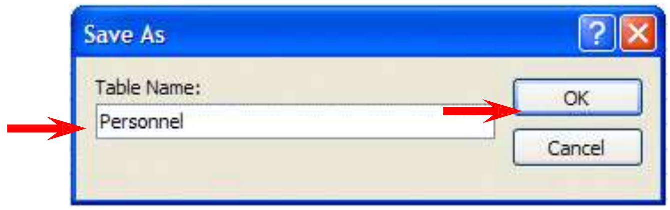
Field Name	Data Type	Size	
Last name	Text	25	(Already Completed)
First name	Text	20	
Social Security#	Text	15	

We'll use an **Input Mask** for our **Social Security Number**. Click in the **Input Mask area** in the **Field Properties** area at the **bottom** of the **screen** (*see left arrow below*).



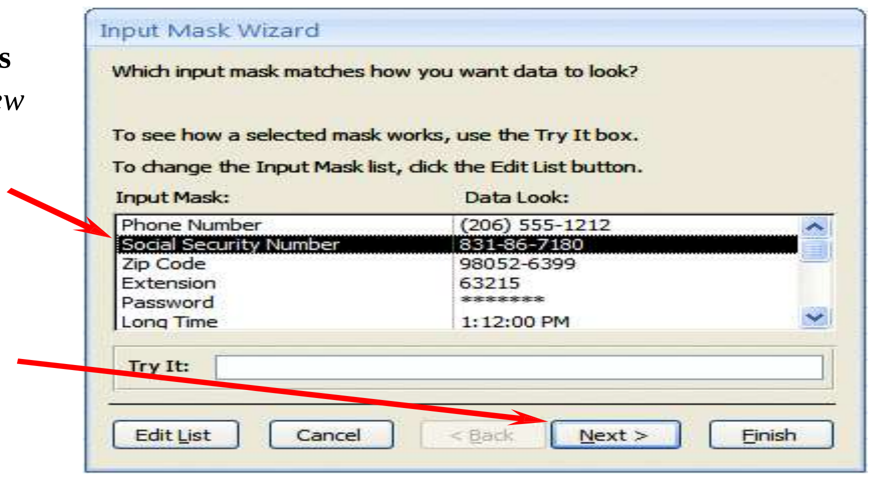
Notice the *three dots (...)* in a box on the right.Click on the three dots (*see right arrow above*).An **Input Mask Wizard** will appear: *Must Save Table First. Save Now?*

Click on **Yes**.

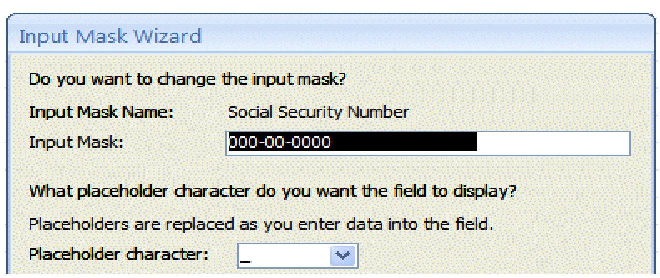


A **Save As** Window may now appear. If it does, type-in **Personnel** in the area under **Table Name** and click on **OK**.

The **Input Mask Wizard** will show you some **Sample Masks** (*you may scroll up/down to view them*).We'll use **Social Security Number**, so click on it.Your screen should look like the one beside.

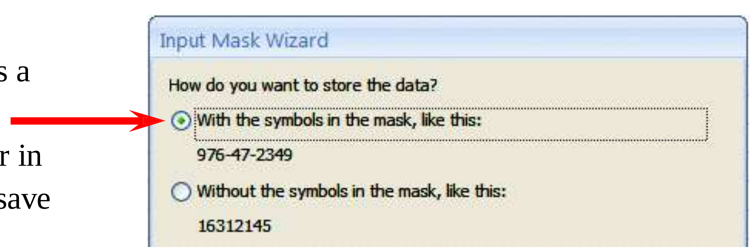


Now click on **Next** at the bottom of the **Input Mask Wizard screen**.
You will now see a default number of **000-00-0000** using **dashes (-)** between the numbers.You can use anything you want.



We'll leave it as it is, so click on **Next** > again (*at the bottom of the Input Mask Wizard screen*).

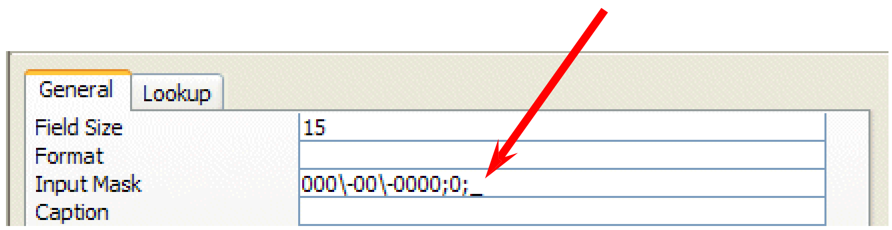
On this Input Mask Wizard screen you'll see *two choices*. Click in the *little circle* to the left of **with symbols in the mask, like this**.
Sometimes, when we use Access data as a part of mail merges or in labels, if we don't save the dashes, they won't appear in our document. So, it a good practice to save dashes.



Then Click on **Next >.**

Now click on **Finish**.

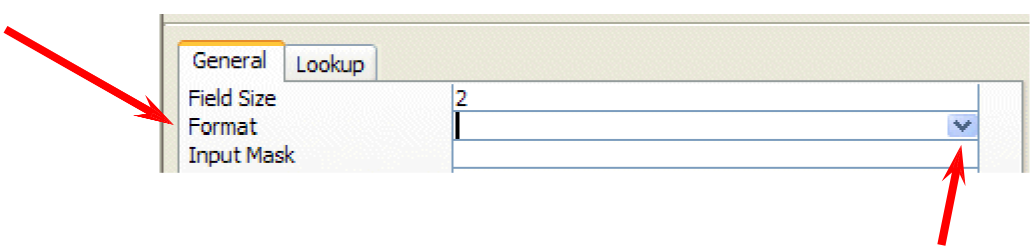
You will see some *special numbers* written in the **Input Mask area** for **Social Security#**. When you begin to enter data in this field, you'll see how this works. Your **Field Properties** area should look like the image below.



Now continue entering the following information in the **Field Name** and **DataType** areas as we did above.

Streetaddress	Text	25
City	Text	20
State	Text	2

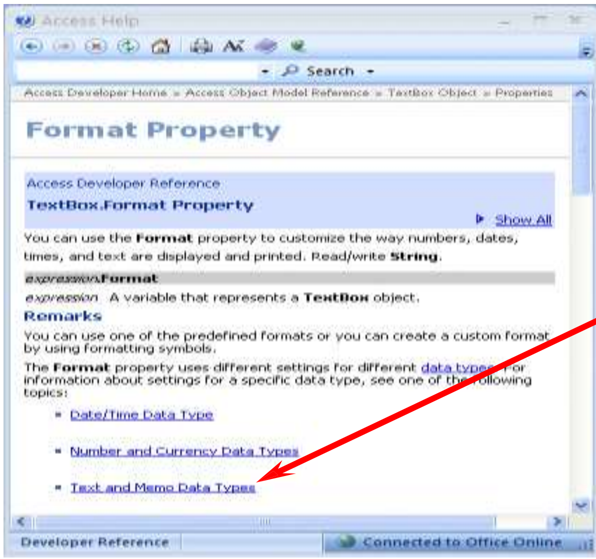
Here we'll use a **Format**. First make the **Field Size 2** then click in the area to the right of **Format**.



A *down pointing arrow* (*like the one above*), will appear on the right side of the **Format** area. If you click on the arrow, the area will appear blank (*that's because we haven't entered a Format*). *Tap* the **F1 key** in the row of Function Keys at the top of the keyboard. A **Help** menu screen (*tailored to Format will appear*) like the one below.

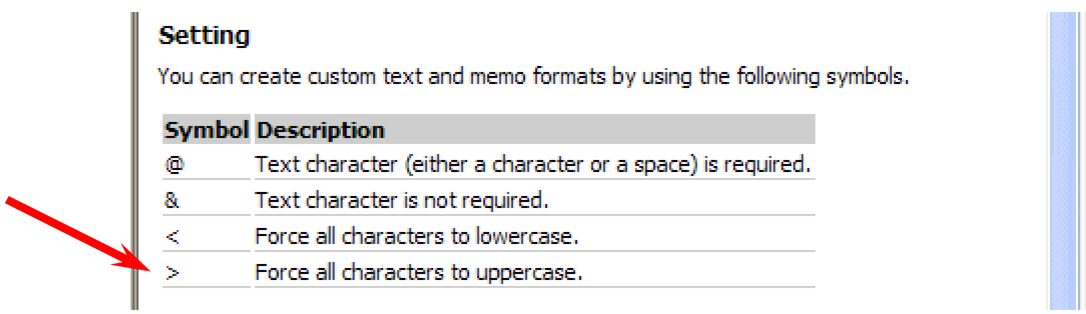
Since you are in the **Format** area, **Help** will be tailored to this area. When the **Help** Window appears, click

Format Property.



Since we are working with a **Text Data Type**, click on **Text and Memo Data Types** (*see arrow beside*).

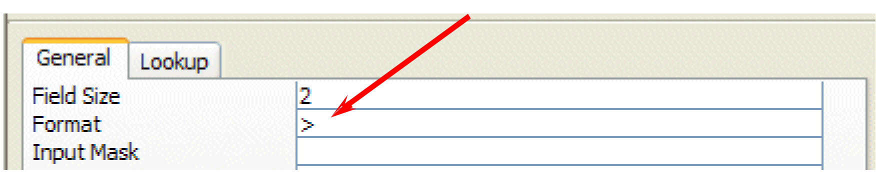
Notice that a > will change any alphabetic character you type into *all uppercase letters*. Now point and click the "**X**" in the upper right hand corner of the Format Help Screen (*notice that the Help Window closes automatically*).


Setting

You can create custom text and memo formats by using the following symbols.

Symbol	Description
@	Text character (either a character or a space) is required.
&	Text character is not required.
<	Force all characters to lowercase.
>	Force all characters to uppercase.

Now type a > in the **Format** area. Your **Field Properties** area should look like the one **below**.



Continue entering the following information in the **Field Name** and **DataType** areas as we did above.

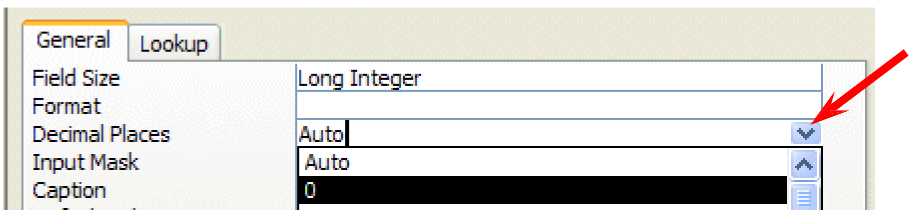
Zip	Text	5
Gender	Text	1

Insert a > in the **Format** area to make *all* **gender** entries become *uppercase* (in capitals – like you just did for **State**).

Favorite Number **Number** (Note: *this is the first number field*)

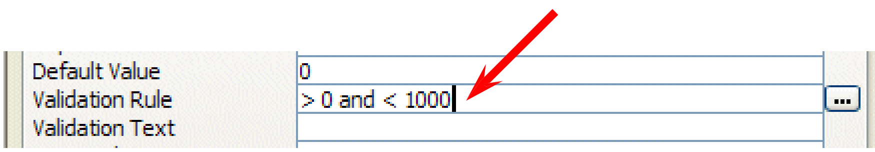
Here we'll learn about **Numbers**, the **Validation Rule** and **Validation Text**. We'll limit the person's favorite number to a number between *1* and *999*. Leave the **Field Size** set to **Long Integer**.

Now click in the area to the right of **Decimal Places**. It currently indicates **Auto**. When you click, you will see a little down arrow on the right side of the area. Click on the little arrow. Select **0**. This indicates that decimal places are not allowed in the **Favorite Number**.



Next, click in the **Validation Rule** area. We'll *build* a *mathematical expression* that will only allow numbers from *1* to *999*. Type-in the following expression (*in the area to the right of Validation Rule*):

> 0 and <1000



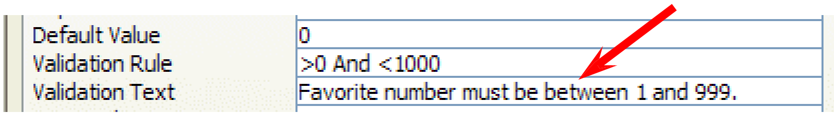
This tells Access that the number entered must be between *1* and *999*.

You'll notice that when you click-in the **Validation Rule** area that *three periods (…)* appear just like they did in InputMask. If you want to click on the three periods they will bring up an Expression Builder which you can use to create the mathematical formula above.

Please note that, If you are not good at math, the Expression Builder can cause problems.

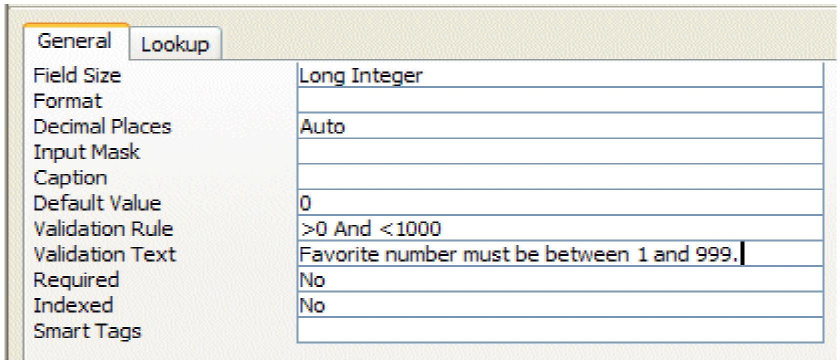
Sometimes, the Expression Builder will *insert* an **<<expr>>** in the formula. If it does this, delete the **<<expr>>**. This will confuse Access, and will frequently cause the program to "*stop*" until you remove **<<expr>>.** So, if you want to look at Expression Builder, please do so but be careful.
If someone does not enter a number correctly, an error message will appear.Now we'll create an appropriate **error message**. **Click** in the **Validation Text** area and **type-in**:

Favorite Number must be between 1and 999.



When you finish all of the above, your Field Properties should look like the one below.

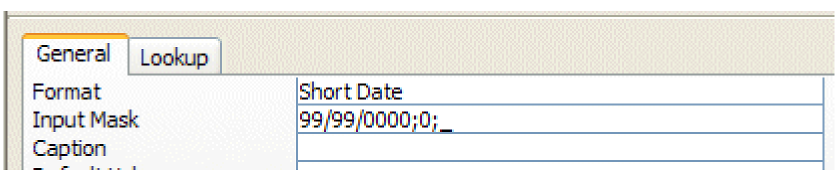
Continue entering the following information in the Field Name and Data Type areas as we did above.



Date hired **Date/Time**

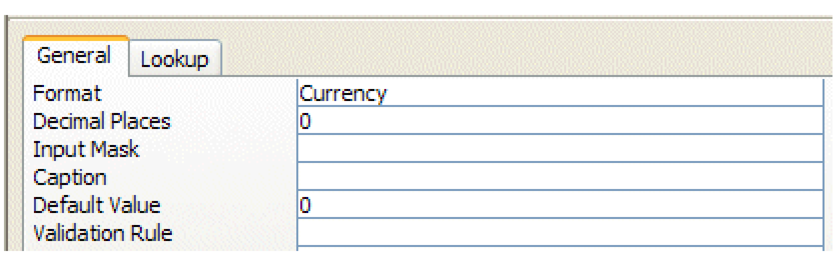
In **Format area** click the **small down arrow** on the right side of the Format area and choose **Short Date**. In the **Input Mask** area **click** the *three dots (...)*, **Save the table** and choose **Short Date** again, click **Next>**, then **Next>** , finally click **Finish**. (*This will insert a / between the day, month, year*).

Your Field Properties should look like the image below.



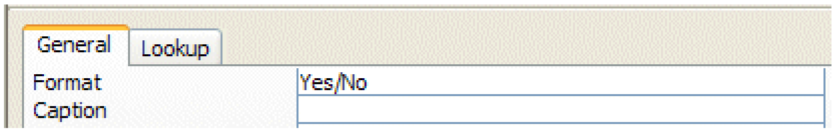
Salary **Currency**

In the **Decimal Places** Field Properties area click on the **small down arrow** on  the **right side** and select **0** – this indicates "no cents." Type a **0 (zero)** in the area to the right of **Default Value.** This will indicate **0** income if no Salary figure is entered. Your Field Properties screen should look like the image below.



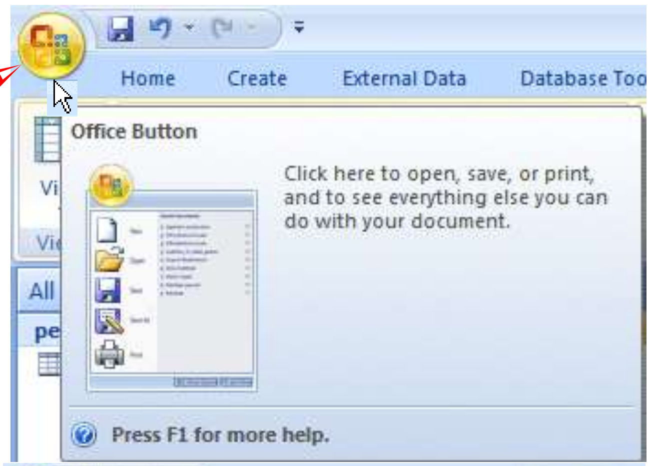
Application Received Yes/No

We'll make this a "**Yes/No**" or "check box" field. When we begin entering data in the database, you'll see how this "box" works.



Now that we have created our Person Database and Personnel Table it would be a good practice to Save the last few changes.

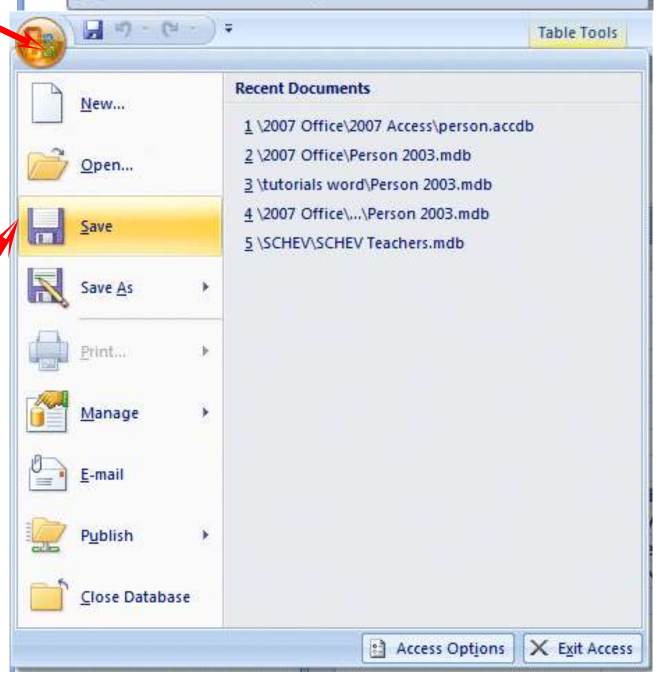
1.4 Microsoft Office Button



The **Microsoft Office Button** has **replaced File** in the Menu Bar. In the **upper left corner** of your Access 2007 screen you will **see** a button similar to the **image** on the **right**. This is the

Microsoft Office Button.

Click the **Microsoft Office Button**.



You will now see the Access 2007

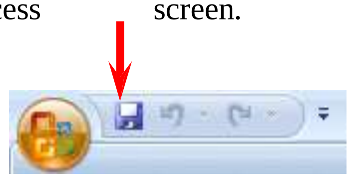
Microsoft Office Button selections.

First, notice that many of the "old" File-Menu Bar choices are included in this menu (they are all here – we'll show you).

Since we Saved our database at the beginning of the tutorial, we only have to click the **Save** choice.

Quick Access Toolbar

You could also click the **small diskette** in the **Quick Access Toolbar** in the upper left corner of your Access screen.



1.5 Entering data in the database

At this point you will still be in the **Design view**. To enter data into your

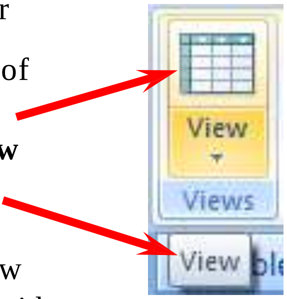


Table you will need to be in a **Datasheet View**. In the upper left corner of your screen (*under the Home Tab*), you will see that the first button on the left that has a small sheet of paper (*see arrow on the right*) – the **View** button. Point to this button with the mouse and pause, you will see a "**Tool Tip**" that indicates that this button is the **View Button**. This is logical because you have been designing your table and now want to view the data that you will enter in your Table (database). If you are familiar with spreadsheets it looks like a tiny version spreadsheet. Click the **View** button.

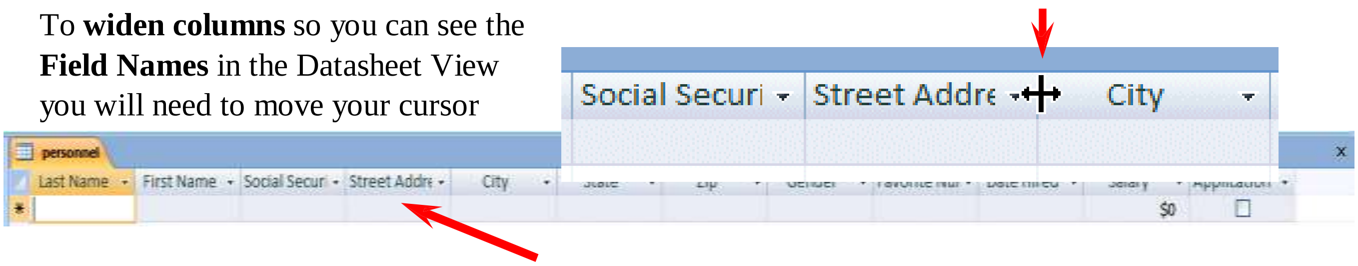
When you click the **View** button, the top of your Access screen will look like the image below.

Notice that all of the Fields you created are in the order you created them along the top of the Table.

Some of the **Field Names** may be "cut-off" a bit. We'll widen our fields in a moment. You can enter data in each Field. But let's widen some of our columns a bit so we can see the titles.

1.5.1 Widening Columns in Datasheet View

To **widen columns** so you can see the **Field Names** in the Datasheet View you will need to move your cursor



line between two Field Names **(**like we did in the image beside**)**. When our cursor was over the line between *Street Address* and *City*, it turned to a line with *two arrows pointing left and right*. When you see this line with the arrows, click and hold down the left mouse button and move your cursor to the right a bit. You will see the column get larger as you move your cursor. When you have the column as wide as you desire, take your finger off the mouse button. You may desire to widen other columns so you can see all of your data – like First Name, Last Name, Street Address, etc. You widen as you desire.

*Note that since we have a **Validation Rule** in one of our Fields, if we try to widen a column we will get the **Validation Rule error message**. So, it's best to only widen columns when you are not entering data.*

1.5.2 Entering Data

Under **Last Name** you will see a *flashing cursor*; this means that you are ready to begin entering data. You may type the data and tap **Enter,** or click with the mouse in each field.

If you make a mistake you may **retype** the data. If you see a mistake later you can come back at any time and correct it.

Under each field, **type the following in the area below the Field Name**:

Field Name	To be typed
1. Last Name	**Smith**
2. First Name	**Fred**
3. Social Security #	**123-45-6789**
4. Street address	**100 Main Street**
5. City	**Lynchburg**
6. State	**Va**
7. Zip	**24501**
8. Gender	**m** or **f** (your choice)
9. Favorite Number	**2007**
10. Date Hired	**07/01/1993**
11. Salary	**40000**
12. Application Received	**Point** the mouse to the little square and click the left mouse button. You will see a **check mark** appear in the square. A click in the square indicates that the application has been received. If you do not click, then that will mean the application has not been received.

As you are entering this data you will notice several things.

Social Security Number and **Date Hired** – You'll see your **Input Mask** work.

State and **Gender** – you typed in small letters – notice how the **Format** (>) forced the Letter (s) to be capitals.

Favorite Number – since the **Favorite Number** is too big, you will see your error message appear. Click on OK in the message screen and then create a Favorite number that will work.

Salary - notice how your **Currency** formatting created a **$**, **commas** and **periods**.

When you have completed typing the information, tap Enter so the cursor will move down to the next record. You are now ready to insert your second entry.

Note: *When you tapped Enter, Access* ***automatically saved*** *your first record. This can be confirmed by the display of the hourglass.*

Also note that as you began typing your first record a small pencil appeared in the left margin. This indicates that you are "writing to" this record (editing). Below the pencil an ***** *(asterisk) also appeared. This indicates that your next record will go below the first.*

There are (**2**) **methods** for **entering data into the database**:

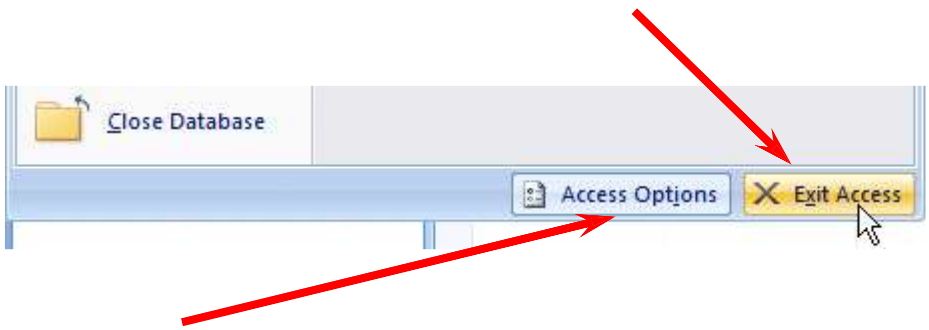
1. The method you just used is called **Datasheet View** method.

or

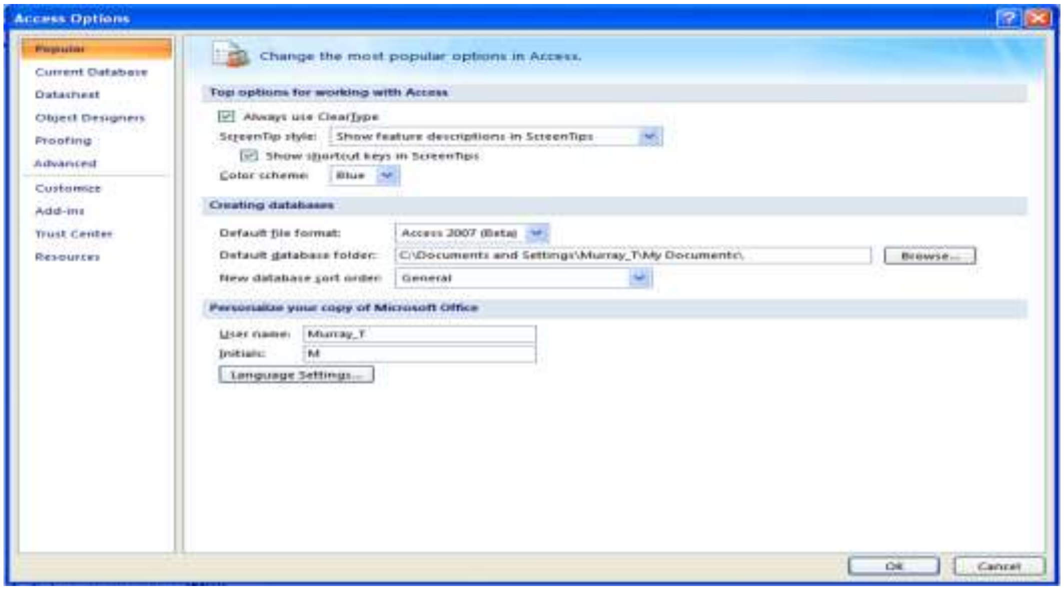
2. You can use the **Form View** method **(**we'll create a Form in a bit later in the tutorial**)**.

1.5.3 Exiting and Saving

Anytime you need to leave your database, click the Microsoft Office Button in the upper left corner of your Access screen, then click **Exit** Access – in the lower right corner of the menu screen. If you have not saved your spreadsheet, a reminder box will appear asking you to do so.

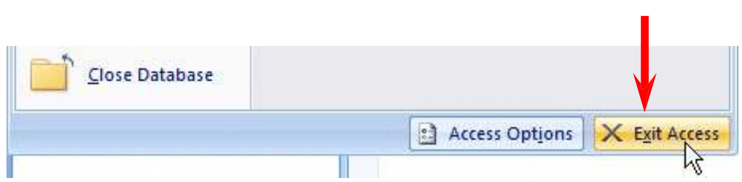


Notice the **Access Options button** to the left of **Exit Access**. Earlier, we indicated that all of the choices under File in the Menu Bar are still available using the Microsoft Office Button. Click the **Access Options button.** The **Access Options menu screen** (*top of next page*) will appear. As you can see, all of the choices available under File in the menu bar are here – as well as many more.



If you click the **Resources** selection in the Access Options menu, you will see some great on-line resources available to assist you with Access.

After you have reviewed the Access Microsoft Office button choices, click the **Exit Access** button**.**



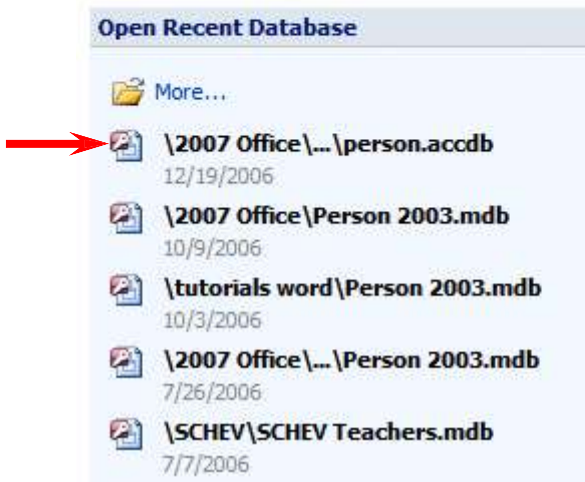
If Access asks, "**Do you want to save?**" click Ok. Incase Access shows you a **Save file screen**, give it a name of your choice and click on **OK**. You should then exit to the Windows Screen with no problems.

Since you have already named everything for this exercise, you should not have to name any files as you exit.

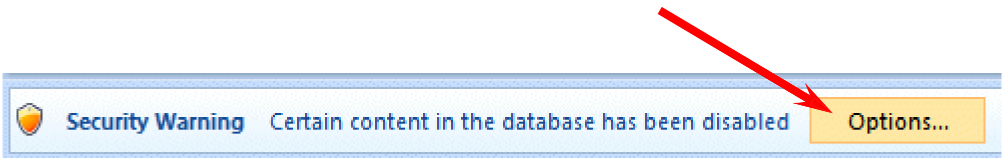
1.5.4 Opening Access Database - again

If you decide to Exit Access 2007, and then return to continue the tutorial, refer to the instructions at the beginning of this tutorial (Page 1) to open Access again.

One good aspect about Access 2007, is the **Open Recent Database** area on the right side of the screen. Once you have created a database, you will see your database in the **Open** portion of the area (*see arrow and image to the right*). You can simply click on the file in this tutorial **Person**, and it will open.

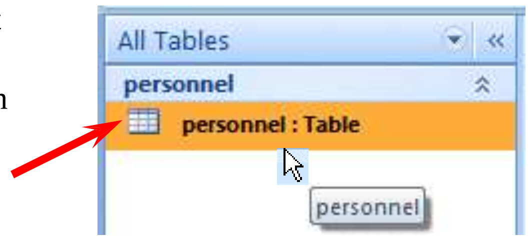


When Access 2007 opens, you will see that the bottom portion of your screen is blue and that the below message appears above the blue area. Access has long been a source for worms and viruses when e-mailed to other people, so Access 2007 has this additional feature to protect you. Click the **Options** button to the right of the Security Warning.



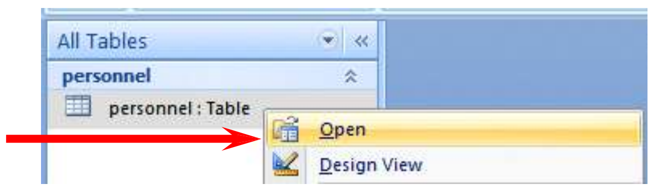
When you click the **Options** button to the right of the Security Warning, the Microsoft Office Security Options menu screen will appear. Take a few minutes to read this information. If you receive an Access database from someone who you do not know, you can use the features in this screen to assist in protecting you. Since you are opening your own database, click the small circle to the left of Enable this content (*see arrow in image below*). Then click the OK button.

When you click the **OK** button you will see, on the left side of your screen, that your **Personnel: Table** turns **orange**. This means that it is OK and that you now can open it again to enter more data. You have **two** choices to open your Table.



1. You can move your cursor over the orange area and click the **LEFT** mouse button **twice** quickly and your Table will open in the Datasheet View.

2. You can move your cursor over the orange area and click the RIGHT mouse button. You will see a *drop down* menu appear. You can click on **Open**



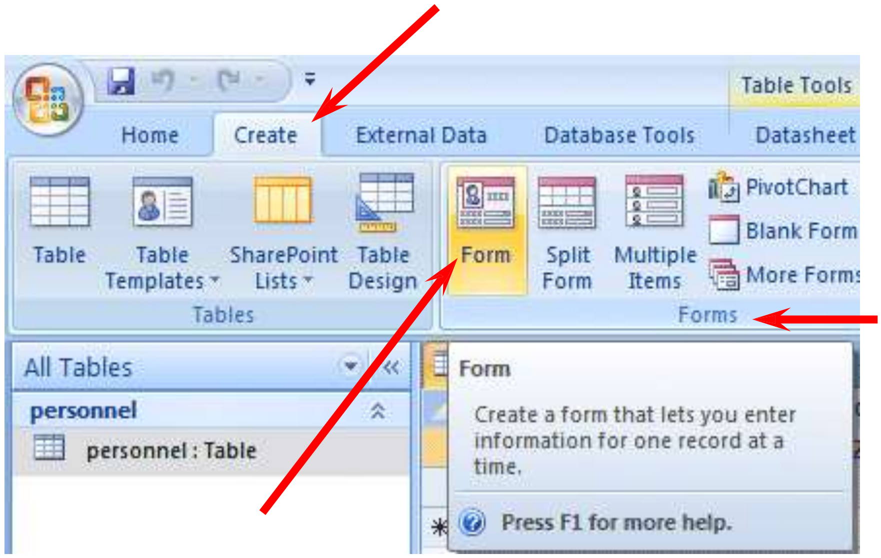
You are now ready to continue entering the data in the **Datasheet View**.

1.6 Form View and Datasheet View

As indicated earlier, you have a choice to enter your data. You can use the **Datasheet View**, like we have done so far, or you can use a **Form**. Both methods work well.

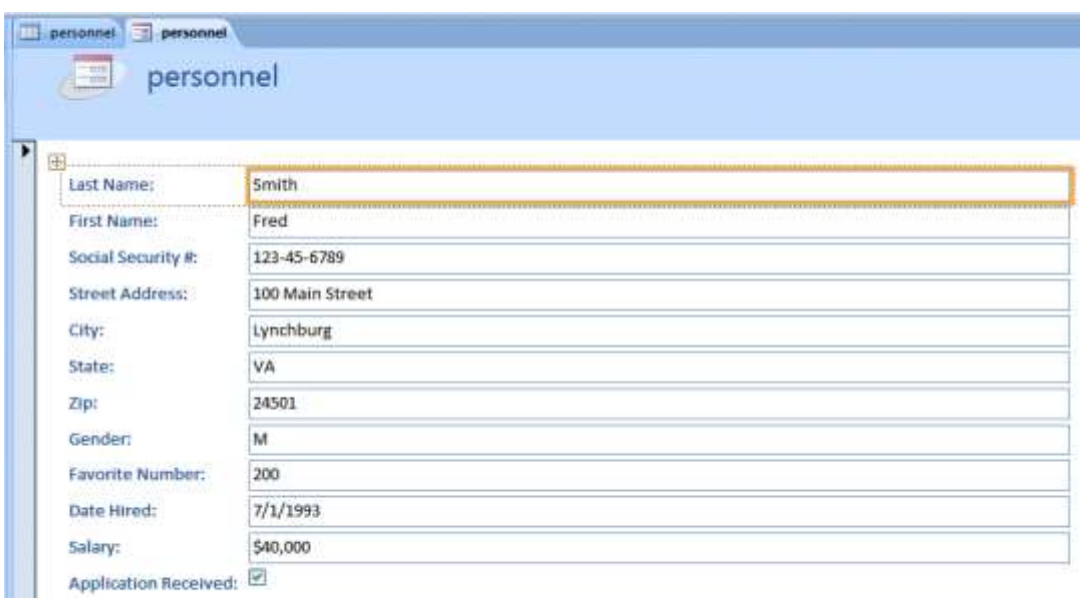
Creating a **Form** in Access 2007 is different in previous versions of Access. We'll now use the new Tabs and Ribbons to Create a Form for our Personnel Table.

First look at the top of your Access screen and click the **Create Tab** (*we want to create a Form*). Look in the **Create Ribbon** in the **forms Group** and you'll see the Form button. Click the **Form** button.

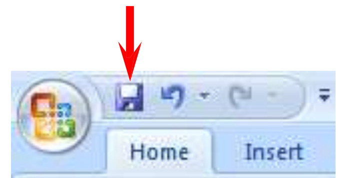


You'll see your screen change significantly in many ways. The first thing you'll notice is that Access 2007 (*knowing you were entering data from the Personnel Table*) created a **Form** with all of the Fields in your Table**.** You're now ready to enter data in this form.

Your **Personnel Form** should **look** something like the image **below**.

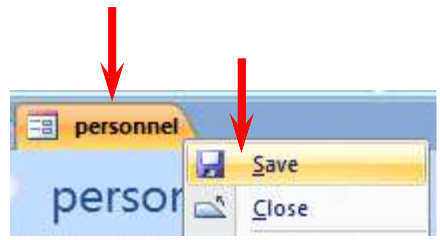


Another feature of Access 2007 is the Tab **layout** for what you're working with. Look just above the **Form** and you will see **two Tabs**. *Take note of the small image on the left of each Tab*. The left Tab is your **Personnel Datasheet** and the right Tab is your **Personnel Form**. You can click on whichever Tab you want to enter Data in your Table.

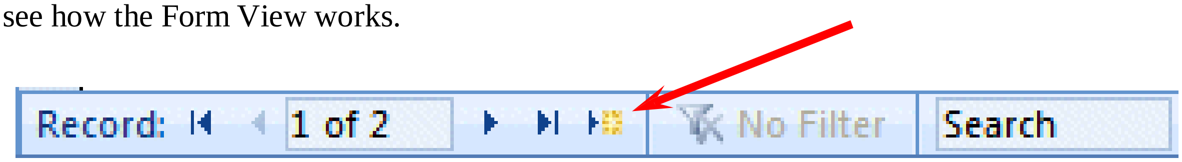


You may enter data in **Form View** the same as in **Datasheet View**. To **Save** your new **Form** you can click on the **Small Save Diskette** in the **Quick Access Toolbar**. Since your **form** will be a part of your **Person Database**, it will automatically save as the **Tab Name**.

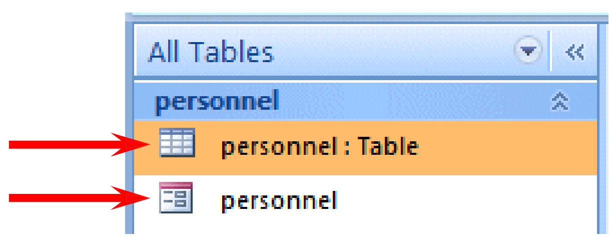
Or, you can move your cursor over the **Personnel Form Tab** and click the **RIGHT mouse button**. When the drop down menu appears, click the **Save** selection.



The data entry **form** is now saved as **Personnel**, just like the Table. Take note, at the bottom of the **Form** screen, there is a **status** area (*see image below*) that tells you what record you are on. You can use the arrows to "move" from one record to another, or select a new record in which to enter data. Click on each of the arrows to see how they work. Some will take you forward or back to the next or previous record, and some will take you to the beginning or end of your records. The **arrow** with an *asterisk* will take you to a **new blank record**. Enter a few records to see how the Form View works.

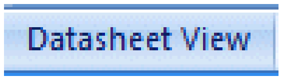


When you first **"open"** your **Person Database**, you may **choose your favorite** method to enter data: The data **Form** or **Datasheet**. Look on the **left side** of your screen and you will see **All Tables**. Under All Tables, you will see your **Personnel Table** with *two selections* below it: **Personnel: Table** and **Personnel.**

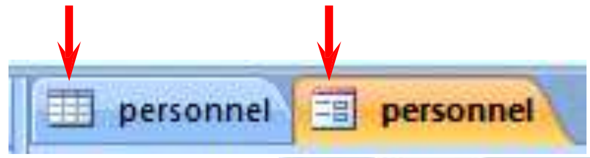


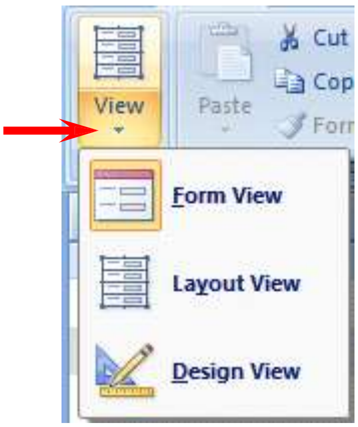
Take note of the image on the left of each selection. You can *switch back* and *forth* from the **Datasheet entry** to the **Form entry** by clicking on **your choice** (*as outlined earlier*).

At the lower left corner of the menu screen, you will see some text that indicates that you are either using **Form** or **Datasheet View** to enter your data.

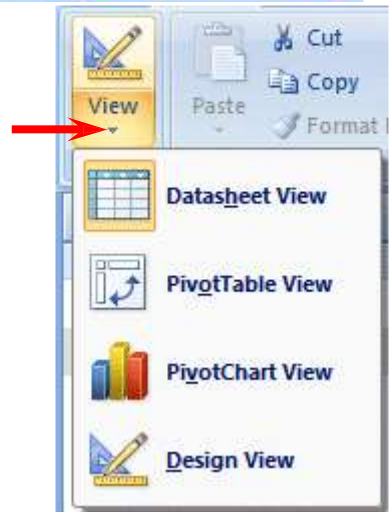


When you have opened your **Datasheet View** and **Form View** you can switch back and forth by clicking on the Tabs.





The **View** button on the upper left indicate what "*view*" you are using: **Design View, Form View,** or **Datasheet View**. You can move back-and-forth between views by clicking-on the down triangle (*see arrows and images*) and then choose the View you desire.



Note: *When you are finished entering data and preparing to exit Microsoft Access, or Close the form, if you did not save before exiting, the program will ask if you want to* ***Save*** *the* ***Form****. This is optional. You may save it with your choice of names and it will then show-up as a form when the Person Database Main Window appears. Or, you can indicate No, and re-create the form again with the Wizard.*

Important!!!

To record enough information so that you can see the power of Access database, which does enter 24 or more records now. You may use either ***Form View*** *or* ***Datasheet View****.*

1.7 Querying the Database

This is what a database is designed for: *finding specific information* about some of the data in the **table**(s) very quickly.

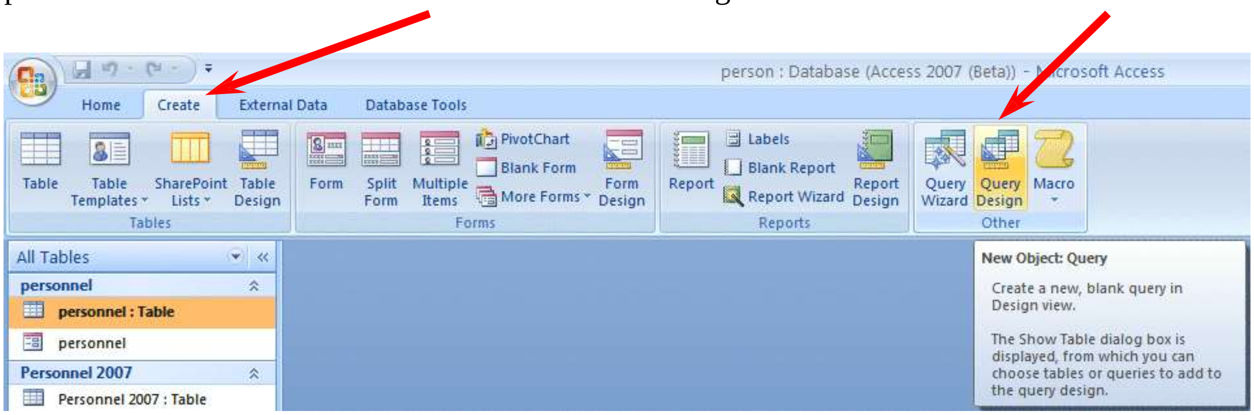
A query is a search for general or specific data in a field or fields in your database (e.g. the first and last names and birth dates of all employees, just the Jones's, the people from CA, salaries > $10,000, etc.).

In order to do this, we need to click on the fields we want to query. So, let's start by finding just **Last Names** in our **table.**

If you are **not** in the **Database: PERSON** screen which **shows** the **Tables, Queries, etc**., go there by **following the instructions on Pages 21-24 .**

If you have the **Personnel Datasheet** or **Personnel Form open** (*to add data*), close them before you begin your queries. The **Access** program sometimes becomes logically **confused** when you try to do queries when it "thinks" you also want to add data. You may see "error" messages if you leave the Form or Datasheet open.

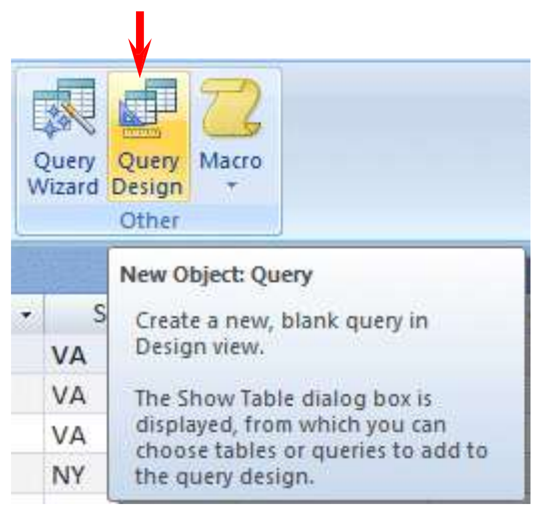
You can right click on the **Datasheet** and **Form Tabs** and then select **Close**, if you have them open. Your Access screen should look similar to the image below.



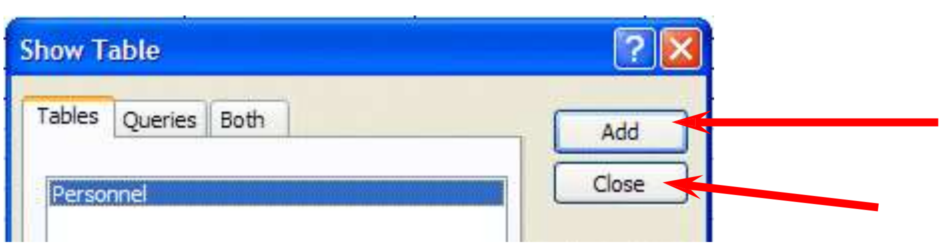
Click the **Create Tab** at the **top** of the Access screen (*left arrow above*). Then move your cursor over the **Query Design Button** in the **Create Tab/Ribbon** (*right arrow above*).

When you move your cursor over Query Design in the Other Group you will see an image similar to the one on the right.

Click the **Query Design button**.



Two new windows will appear: **Query 1: Tab** and **Show Table**. You will **first** have to select the table(s) you desire to query. The **Show Table screen** should look like the one below.



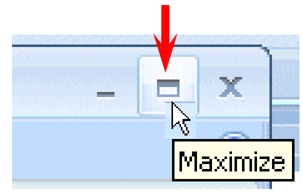
Click **Add First**

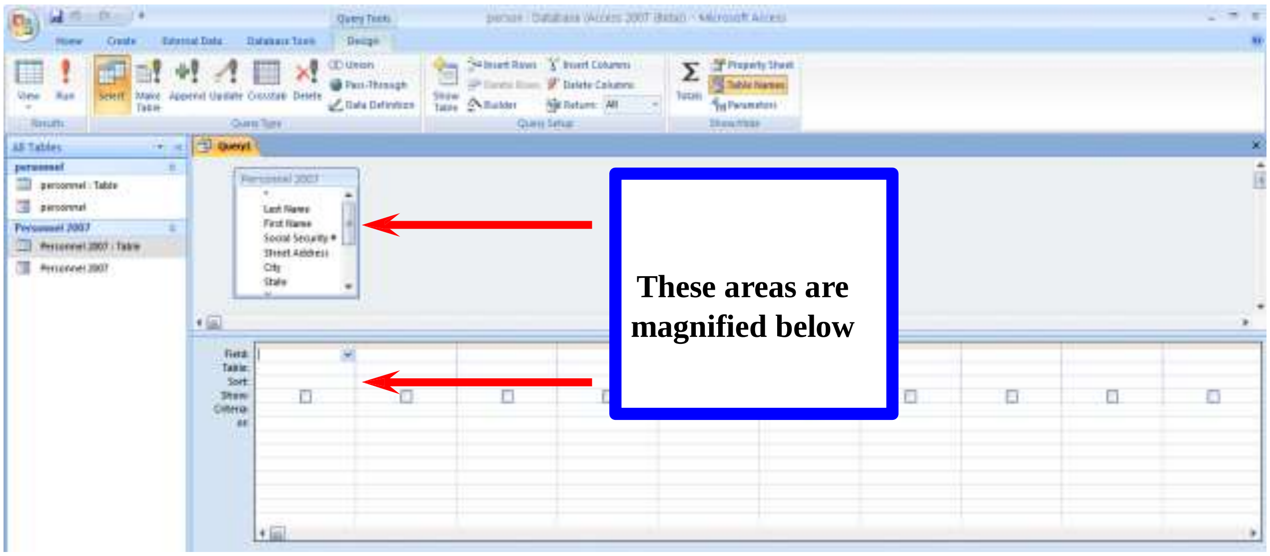
Then click **Close**

The **Show Table** window will disappear, and the **Query 1: Query** window, will appear.

Click on the **expansion square** in the upper right corner to enlarge the **Query 1: Select Query** window.

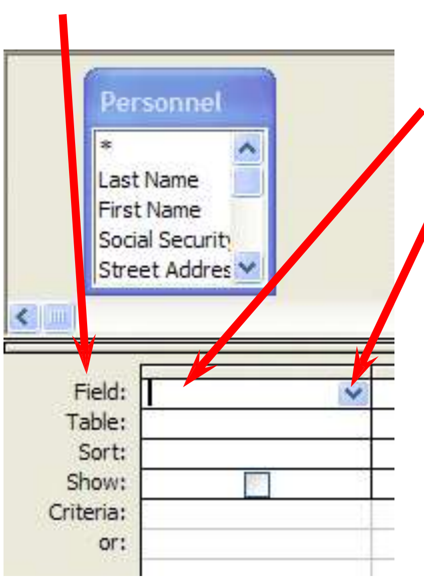
Your screen should now look similar to the one below.





Notice, in the upper half of the window**,** a small box on the left indicates: **Personnel**. At the top is an asterisk (*) and below, in an *elevator* box, are the fields from the **Personnel Table** (*you can move up-and-down the list as you desire*).

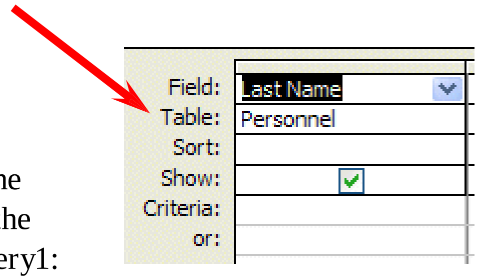
What we need to do next is place the Fields we want to query in the lower area of the screen. Take note of the lower area on the left border. The first row indicates **Field:** followed by **Table, Sort:, Show:, Criteria:,** and **or:**.



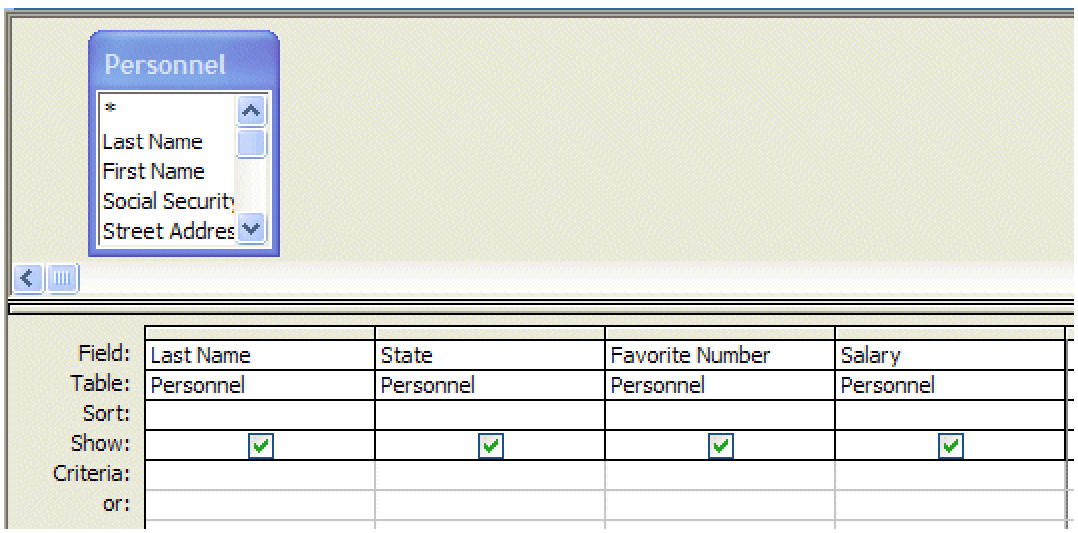
In the lower half of the screen click in the first cell to the right of **Field:**. We'll start with a query on **Last Name, State, Favorite Number** and **Salary**. Now click on the **down arrow** and then click on **Last Name**. **Notice** how **Last**

Name now appears to the right of **Field**: and a  (*check*) is seen in the **Show**: cell (*The*  *means that you will see **Last Names** in your query*.). Take note of the right area of **Table**, that **Personnel** (*the Table from which we queried) is showing.*)

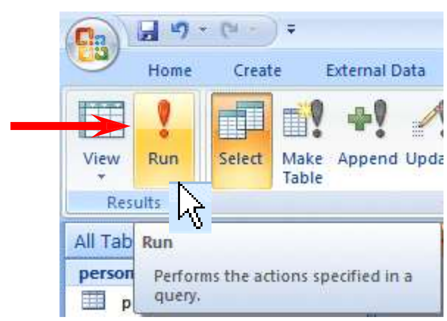
Your **query screen** should now look like the one on the right.



Now move to the next Field cell on the right. Using the down arrow click on **State**. In the next two fields to the right, insert **Favorite Number** and **Salary**. Your Query1: Select Query screen should look like this:



Now, look in the **Button Bar** at the top left of the **screen**. In the middle of the bar you will see an **exclamation mark** (**!**) like the one on the right. If you move the cursor over it, the help text box will indicate "**Run**." Click on the (**!**). *This click executes your query*.

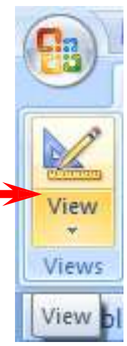


Last Name	State	Favorite Nu	Salary
Sandston	VA	12	$35,000
Kern	VA	200	$100,000
Warren	VA	827	$100,000
Smith	NY	55	$25,200
Warren	VA	8	$45,100
Smith	WV	426	$22,900
James	WV	324	$29,500
Binswager	MO	777	$41,950

Your query screen should look similar to the one on the left.

You will Notice that the screen ONLY shows the four fields that you queried.

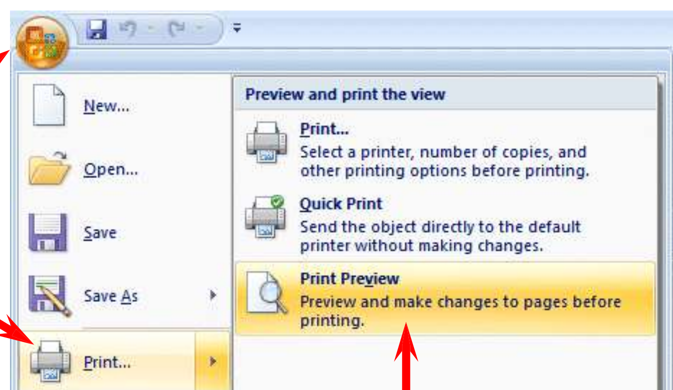
You can **add** or **remove fields,** as you desire. To do this we need to return to the **Design View** where we created this query. To **return** to **Design View** click on the **small button** in the upper left corner of the screen that has the **blue triangle, pencil, and ruler** (*like the one on the right*). Then, simply click in the **Field area** and select a **new field** and it will **replace** the old one. Or, click on the field you want to **remove** and tap the **Delete key**. Sometimes you may have a lot of fields and it will be too large for a single sheet of paper.



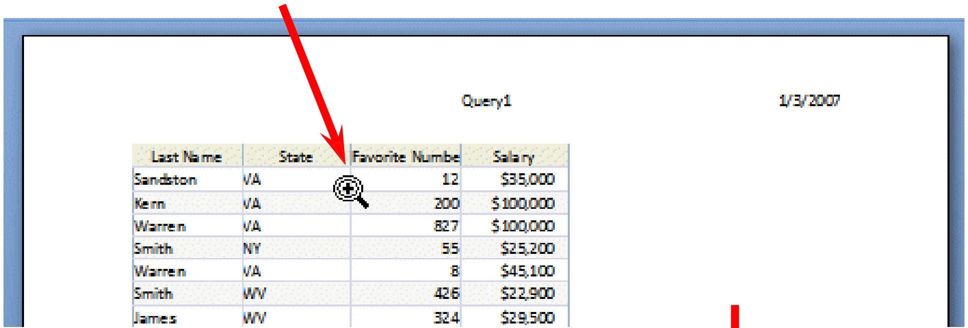
To have preview of your **query**, we'll show you how to open a **Print Preview**

First, click the **Microsoft Office Button** on the upper left of your Access Screen.

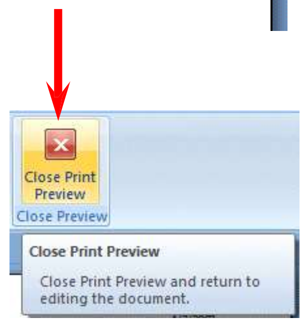
Then move your cursor over the **Print selection**. A **Preview and print the view** area will appear on the right. Move your cursor over **Print Preview** and **click** on this selection.



While you're in the Print Preview you'll see a little magnifying glass that you can move over your query. If you click the left mouse button once the magnifying glass will "*zoom in*" and enlarge the view. If you **click** the left mouse button **again** it will **zoom out**.



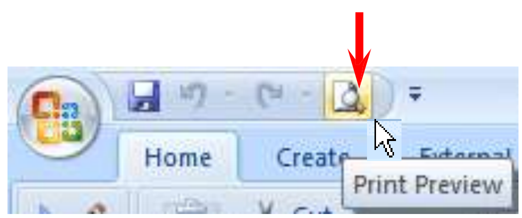
To **return** to your query, click on the **Close Print Preview button on the right side of the Print Preview Tab/Ribbon.** This will take you back to the Normal **View** of your query.



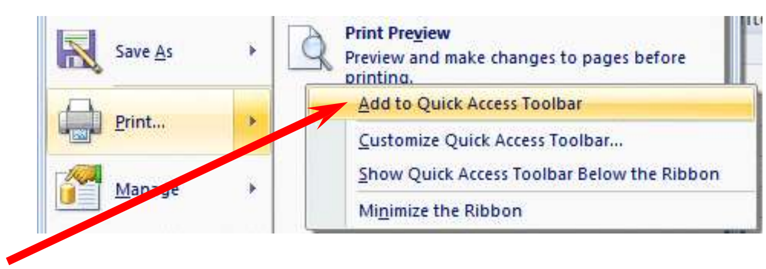
Adding a Print Preview Button to the Quick Access Toolbar.

It took a lot of "*clicks*" to get to Print Preview. So, let's **add** the **Print Preview** button to our **Quick Access Toolbar** at the top of the screen. Then all we'll have to do is click this button for a Print Preview!

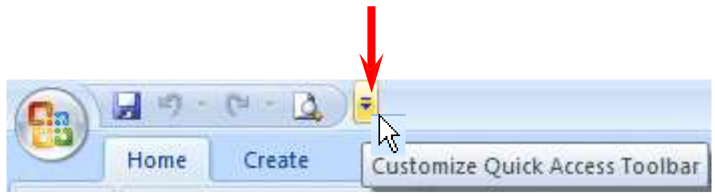
Repeat the steps at the top of this page to "get to" the **Print Preview** selection. When you see the Print Preview selection – **click** the **RIGHT mouse button**. When the drop down menu appears, click the **Add to Quick Access Toolbar** selection.



After you add the Print Preview button to your Quick Access Toolbar, the toolbar will look like the image on the right. Notice that Print Preview may now be accessed by clicking this button.



You can use this method to add any buttons you desire to your Quick Access Toolbar. There is a small arrow on the right side of the **Quick Access Toolbar** (*image on right*). If you move your cursor over the arrow, you will see that it indicates **Customize Quick Access Toolbar**. You can work with this as you desire.

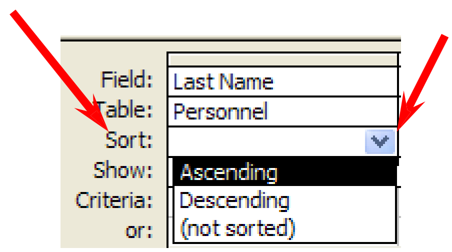


You can customize the Quick Access Toolbar in each 2007 Office application as you wish.

1.8 Sorting the Database

If you are not in the **Query Design Screen**, you'll need to be in that view. So, **go** to the **Design Screen** (bottom Page 28).

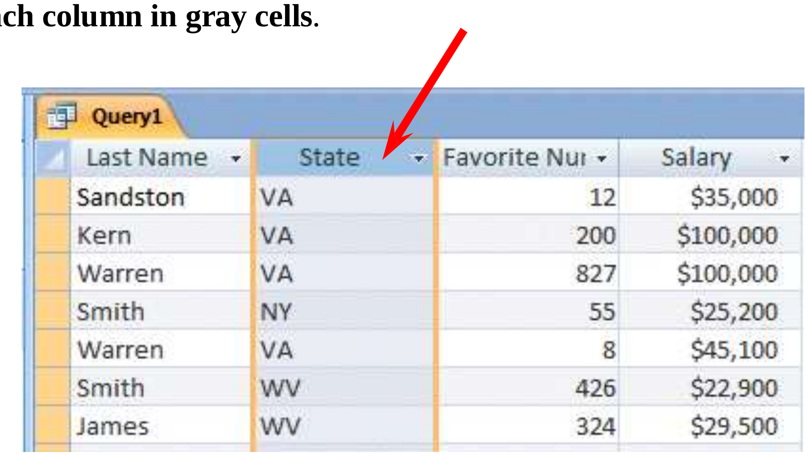
*Note: The third row in the lower half of the screen, indicates **Sort**: (like the image at the right). Click in the **Sort:** area under **Last Name**.*



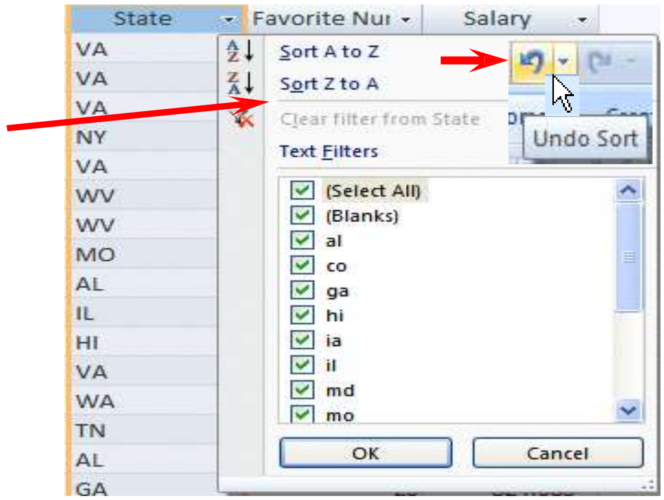
When a down arrow box appears, click on the down arrow. Let's sort the **Last Names** in **ascending** order. Click on **Ascending**. Notice that **Ascending** now appears in the **Sort:** area. Click on the (**!**) to see the **new query**. You would notice that the names you entered are alphabetized. Click on the **Design View button (***triangle-ruler-pencil***)**. Now change the **Ascending** under Last Name to **not sorted.** Now, try sorting some of the other fields on your own. When you are finished remember to set the fields to **not sorted** unless you do want to sort on those fields.

You may also sort various fields in your database whenever you are in the **Datasheet View**, whether you are viewing the entire **Table**, or a **Query** from the Table. Notice that **the Field Names** are shown at the **top of each column in gray cells**.

If you click on one of the blue area field names (*like **State***), the **entire column** (*Field*) turns **blue** (*like the **image** on the **right***). This indicates that you have **marked** the entire column (*Field*).



On the **Home Tab/Ribbon**, on the right side, in the **Sort & Filter** group, you will see two buttons with down arrows (*like the image on the right*). When you move the cursor over these two buttons a text help box will indicate: **Sort Ascending** or **Sort Descending**. If you click on one of the buttons, the **Field** which you selected (*highlighted*) will be sorted in the order selected. Give this a try and see how it works.



You can always click the **Undo Arrow** in your **Quick Access Toolbar** if you desire to go back to the original order.

1.8.1 New to Access 2007 – A Filter/Sort Arrow for each Field!

In the image on the right, we clicked the small down arrow to the right of **State**. A **drop down Filter/Sort menu appeared**!

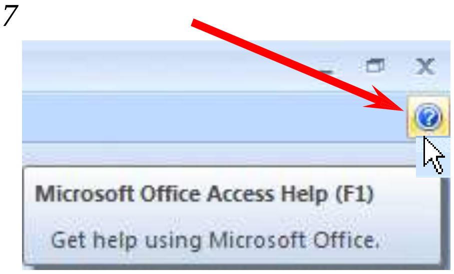
This is new in Access 2007. You also use this menu to enhance your sorts.

Notice that the **Sort A to Z** and **Sort Z to A** are in this menu, as well as text Filters. You can carryout some tests with these filters as you desire. This is a good feature that will assist you greatly in making your queries.

So, there are several ways you can sort your **Tables** and **Queries**.

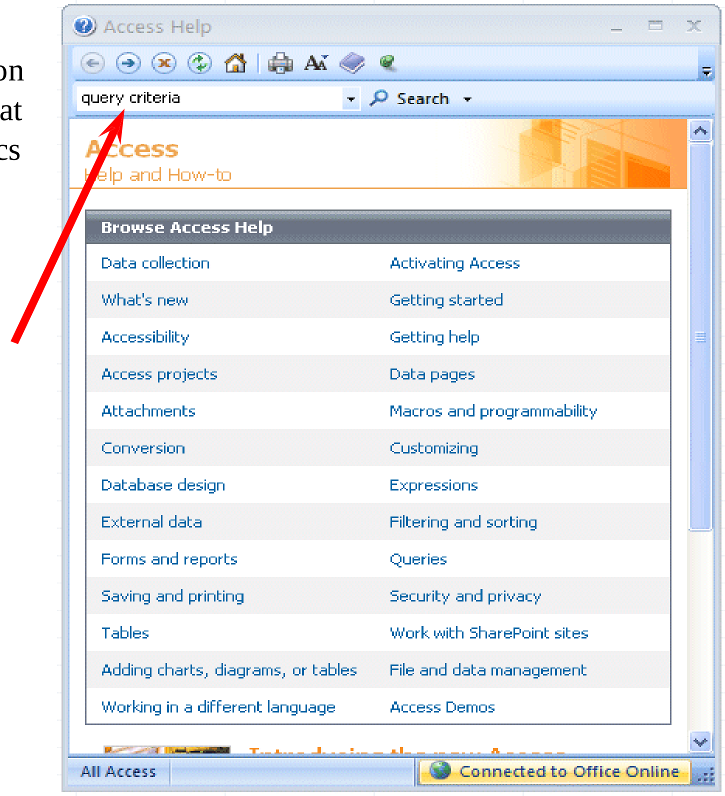
1.8.2 Specific Queries

So far, we have listed everything under each Field Name that we selected. However, many times you will probably want to find something specific in your Table (*database - e.g. people from a certain state or city, people whose favorite number is 7 or salaries between $ 20,000 and $ 50,000*). This is fairly logical, but can be tricky.



To get an idea of various criteria you might want to use, click on the **Help Question Mark** in the upper right corner of the Access Screen

The **Access Help** Task Pane will appear on the right side of your screen. Notice that there are already a number of Help topics already included in this Task Pane.

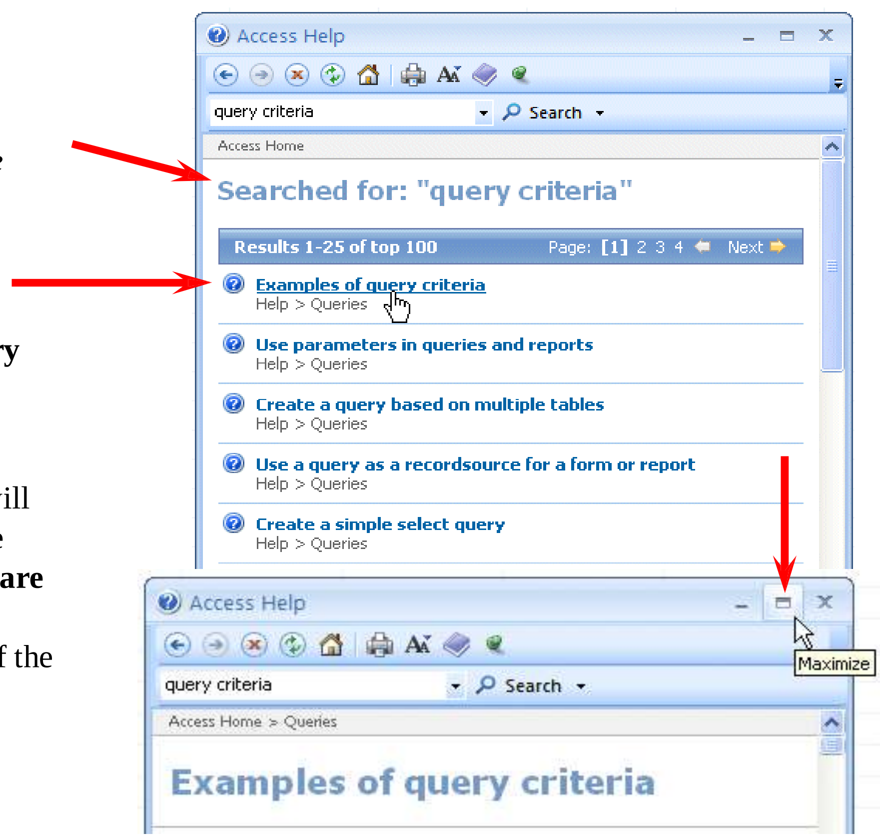


When your screen appears, click in the Search area in the Task Pane. Type-in "*query criteria*" and then tap the **Enter key**.

A *Search for:* Task Pane will now appear (*like the one on the right*).

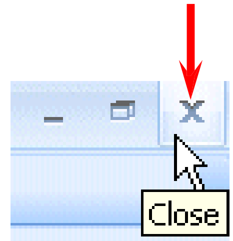
Click on the **Examples of query criteria** selection.

Another **Access Help** Screen will appear similar to the one on the right. Click the **Maximize square** in the upper right corner of this screen so that you can see all of the criteria examples with ease.



Use the **Elevator Bar** on the right side of this **Help screen** to view all of the various criteria for queries. This is a wonderfully updated feature in Access 2007 that really gives you an Access Manual for criteria.

When you are finished, click on the "**X**" in the upper right corner of the **Microsoft Access Help Topics** menu screen to **close** the screen.

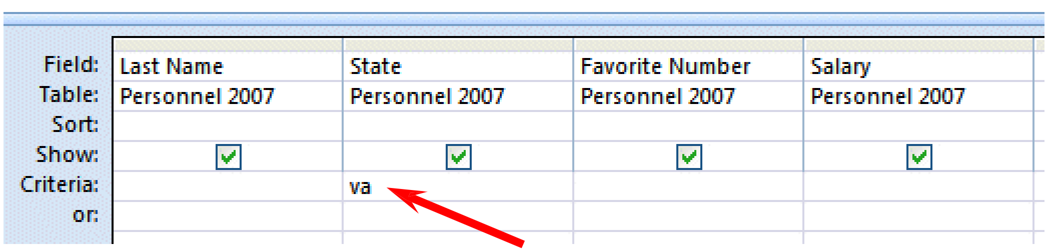


You may return and explore additional help screens as you become more acquainted with Access 2007 database. These help screens are similar to having a complete Access 2007 manual on your computer.

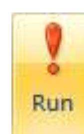
Now we'll try a few **specific queries**. First let's find a **specific state**.

You should be back in the **Query1: Select Query** menu **Design** window. It should look like the image below. If you're not then click the **Design button** in the upper left corner of the screen (*like the one on the right*).

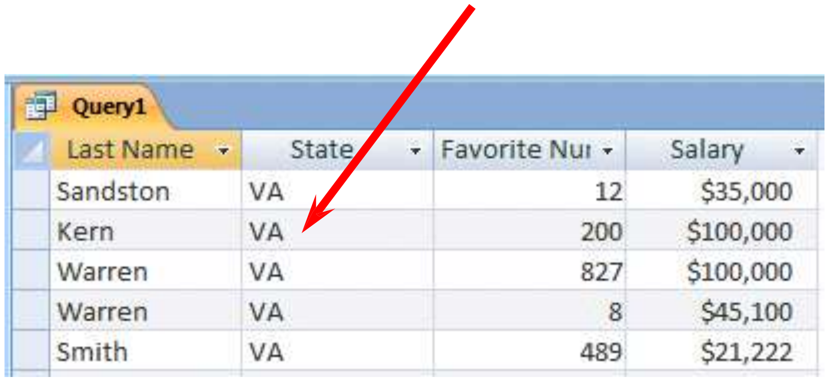
Click in the cell to the right of **Criteria,** in the **State** column. You will see a flashing cursor (*Make sure you are in the **State** column*). Type-in the abbreviation for one of the states you entered in your Personnel Table. Your Query should look like the image below.



Now click on **(!)**. A new Query1 window will appear. ONLY persons from the state you selected should appear.



This is a **specific query** for that **state**.



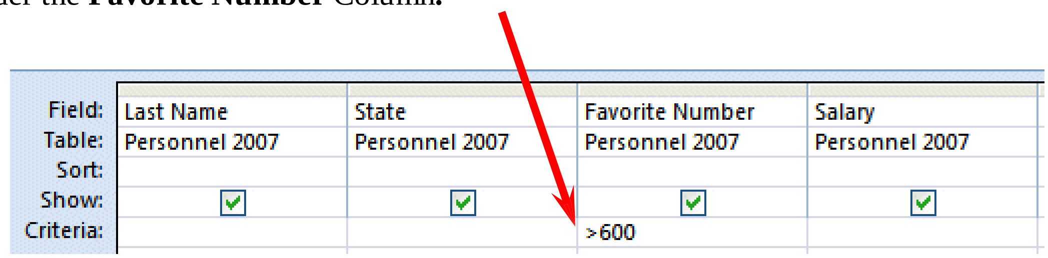
Query1

Last Name	State	Favorite Nui	Salary
Sandston	VA	12	$35,000
Kern	VA	200	$100,000
Warren	VA	827	$100,000
Warren	VA	8	$45,100
Smith	VA	489	$21,222

Click on **Design View Button** (*triangle-ruler-pencil*) to return to Design View.

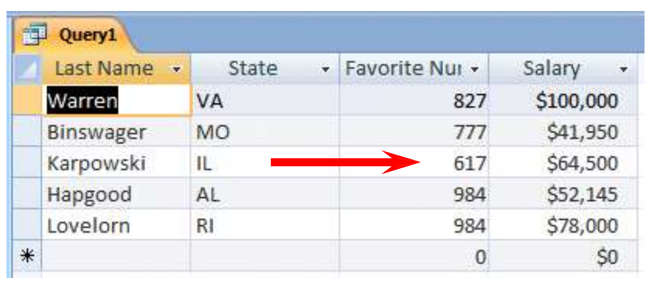
Delete the **state** you entered.

Now we'll look for **Favorite Numbers larger than 600**. Type-in **>600** in the **Criteria** cell under the **Favorite Number** Column**.**



Field:	Last Name	State	Favorite Number	Salary
Table:	Personnel 2007	Personnel 2007	Personnel 2007	Personnel 2007
Sort:				
Show:	☑	☑	☑	☑
Criteria:			>600	

Click on (**!**). Everyone with a favorite number larger than 600 should show. If no one is indicated, this implies that you don't have a person with a **Favorite number larger than 600**, or you might have typed the **>600** incorrectly.



Query1

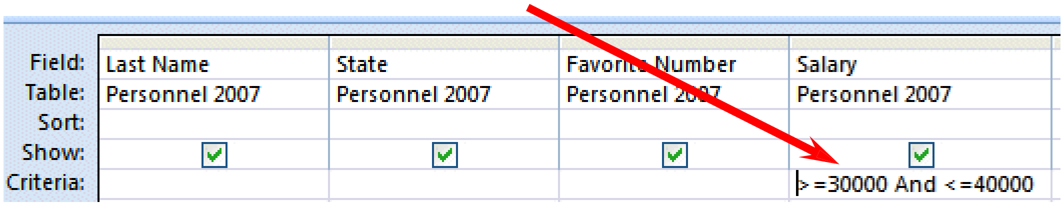
	Last Name	State	Favorite Nui	Salary
	Warren	VA	827	$100,000
	Binswager	MO	777	$41,950
	Karpowski	IL	617	$64,500
	Hapgood	AL	984	$52,145
	Lovelorn	RI	984	$78,000
*			0	$0

Return to the **Design View**. Delete the **>600** and *run the query with no criteria*. You should see all the fields again. Return to the **Design View** again.

Make sure all the Criteria cells are empty!

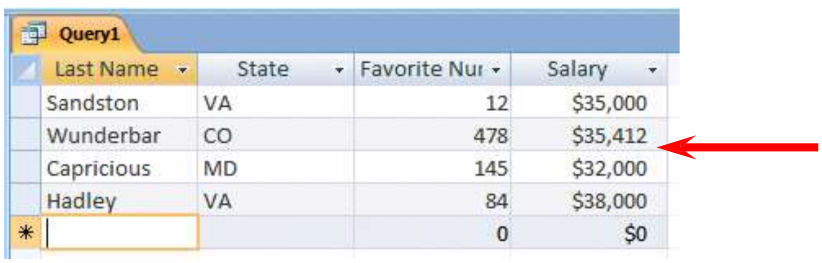
Next we'll look for persons with **salaries equal to or larger than $ 20,000** and **equal to or less than $ 50,000**. In the **Salary** field column, in the Criteria**:** cell **type-in**:

>= 30000 and <= 40000



Click on the (**!**).

You should now see a specific query that indicates those persons in the range you've chosen.



Go back to **Design View**. Delete the criteria you entered under **Salary**.

Now, on your own, if you desire, add or delete some fields to your query and try carryout tests with some combinations.

Don't get frustrated if no specific items appear. Frequently you might query for something that *can't exist* (*e.g. states of **VA** and **CA** – a person can't be from both states at the same time*) or there isn't anything that matches your desired query.

To enhance your skills, notice the **or:** just below Criteria to the left of the **Design View**. Try one state in the **Criteria:** cell under **State** and another in the **or:** cell**.** Go back to your Query Help (*Page 32*) and try some of the criteria you see.

When you have a good feel for queries, you're set to end your query session.

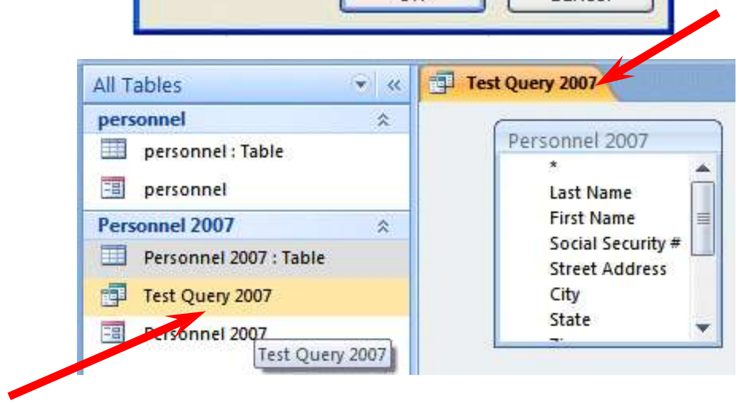
Make sure that your query is completely clear of criteria – we'll use this query later for a report!

Right click on the **Query1** Tab**.** A Drop down menu will appear. Move your cursor over **Save** and click on **Save**.

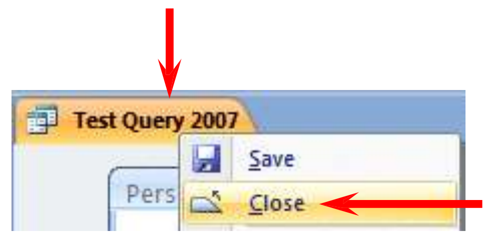
A **Save As** window will appear. Give the **Query** any name you desire. In this tutorial, we're going to name our query **"***Test Query 2007*". Click on **OK**.

When you click the **OK** button you will notice several things - the **Query1** Tab changes to the name you've chosen and on the left side of your screen, you can see your new query available for use again as desired. You can activate this query and change things just like you did above

You can **close** your **Query** by right clicking on the Tab and then clicking **Close**.



Any time you desire to **open** this **query** again, you simply click twice quickly on the query selection on the left side of your Access screen. Or, you can right click the **query** and choose **Open**.



1.9 Reports

Reports can be very complex. In this tutorial we'll cover the basic steps of creating Reports. *A good manual or some knowledgeable assistance will be essential for mastering reports.*

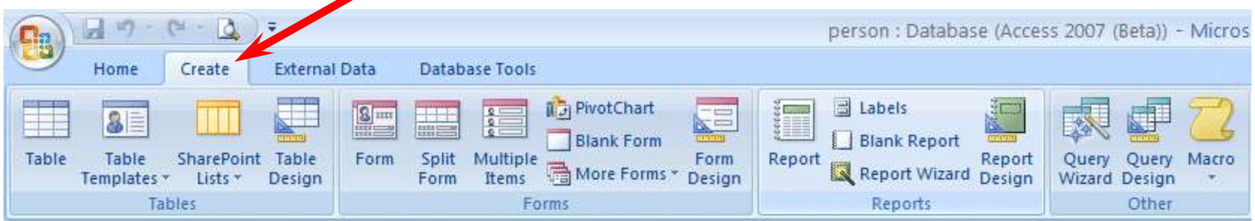
There are several types of reports. We'll use Access **Wizards** and Tabs/Ribbons to design several simple reports.

For those who are acquainted with Access reports from earlier versions of Access – this will be a whole new adventure with the Tab/Ribbons of 2007 Office.

First, make sure that you have closed any Tables, Forms, or Queries which are open.

Your Access screen should look like to the image below.

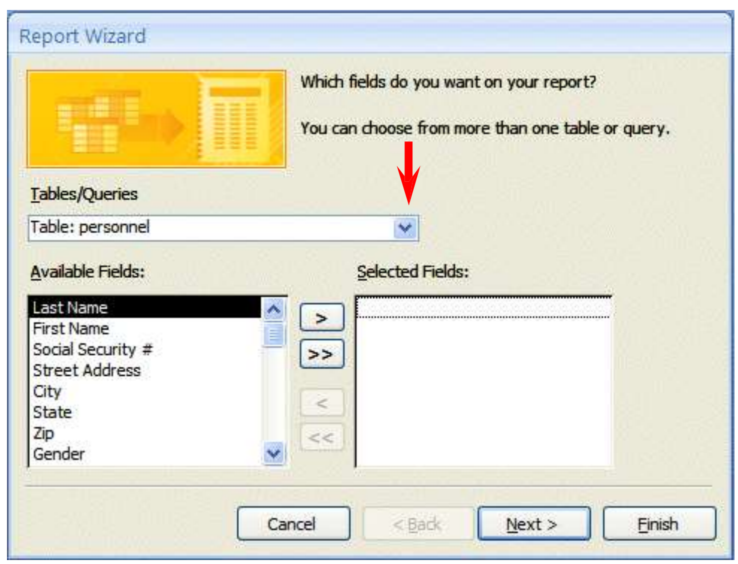
Click the **Create Tab** and the **Create Ribbon** will appear. Look at the Ribbon on the right side. You will see the **Create Reports** Group



The **Create Reports** Group is enlarged on the right. We'll begin our Report lesson using the **Report Wizard**. Once you've created a Report with the Wizard, you'll be a little familiar with Reports. We'll then use some of the other selections in this Group to create Reports.

Click the **Report Wizard** selection in the **Reports Group**.

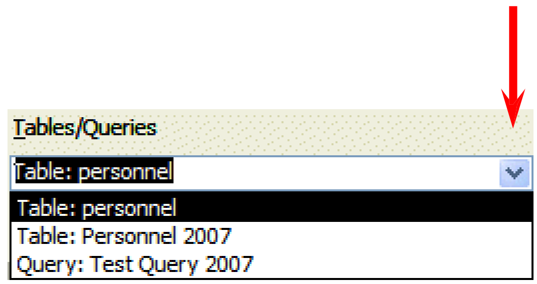
The **Report Wizard** Menu screen will appear.



Read all the information in the **Report Wizard** menu screen. Only the fields you select from your table will show in the **report**. To bring fields into the report individually, you should click on the **name** of the **field** (*in the list of fields in the area under **Available Fields***) and then click on the >. The order which you use in selecting the fields, will be their respective order (*positions*) in the report. The >> imports all the fields. The < exports the fields which you have selected and <<

exports all the fields. Incase you make a mistake or want to start over, click the << selection to undo all the fields and try again.

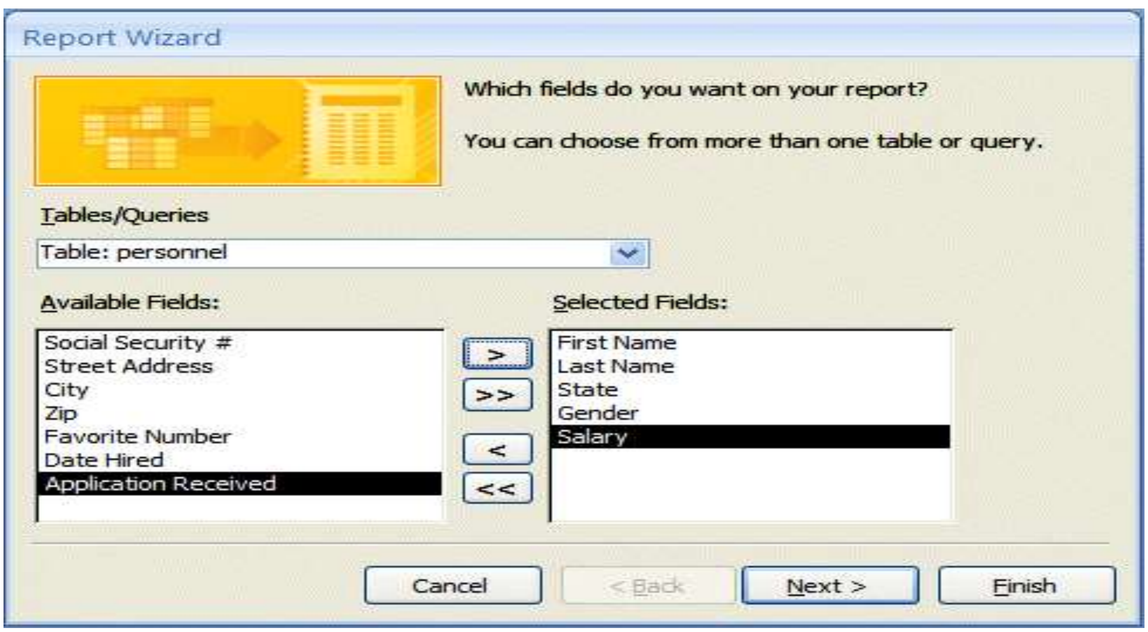
Look at the image beside. The arrow pointing the down on the right side of the **Tables/Queries** area. Click this arrow and select your **Table: Personnel**.



Your **Report Wizard** menu screen should look like the large image in the middle of the next page

Let's begin. Click on **First Name**, then click on > (*take note of how the First Name field moved from the* ***Available Fields*** *to* ***Selected Fields***). Now do the same with the **Last Name, State, Gender** and **Salary** fields. These are the fields that will appear in our first report.

Your Report Wizard menu screen should look like the one below.



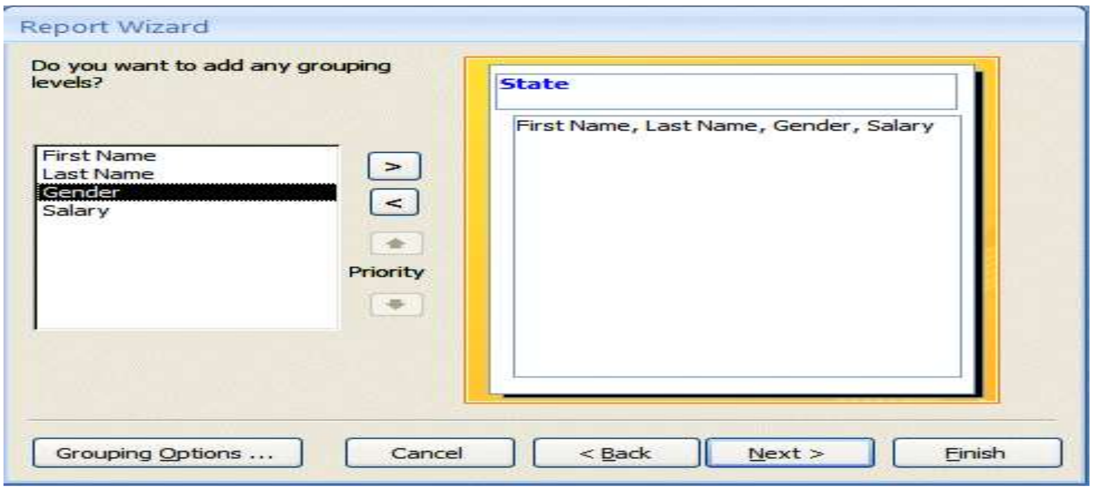
If it does, click on the **Next** > Button, else use the << to undo all the selected fields and try again.

1.9.1 Grouping in Reports

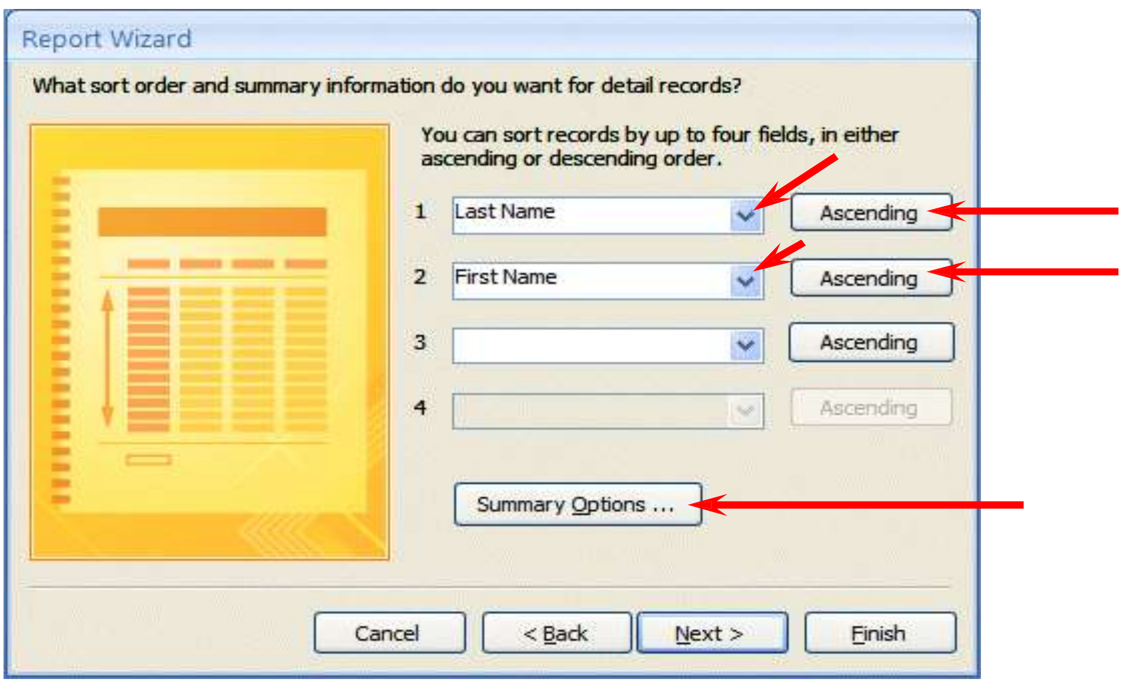
This **Report Wizard** menu screen asks if you want to add **Grouping**. Grouping simply "*groups*" records by an item in the report you are designing. We'll group by **state**. This means that records from a state will be in a "*group*" (*e.g. people from Virginia will be in one group, the folks from*

Washington in another, and so on). This will be easy to see when we look at the report. So, click on **State**, then click on >. If you make a mistake, no worries, just use the < selection.

Your screen should now look like the one on the following page.



Click on **Next >** again. Another **Report Wizard** menu screen will follow.

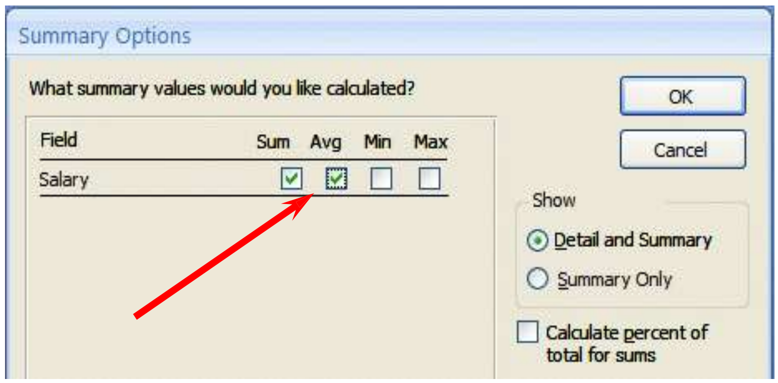


First, the above screen requests that you indicate a **Sort Order**. This simply means that within each group, the **alphabetic order** in which you want the fields sorted. We'll sort by **Last Name** and then **First Name**. This way you'll have the names, grouped by state, in Last Name order and where you have several people with the same Last Name, they'll be sub-sorted in First

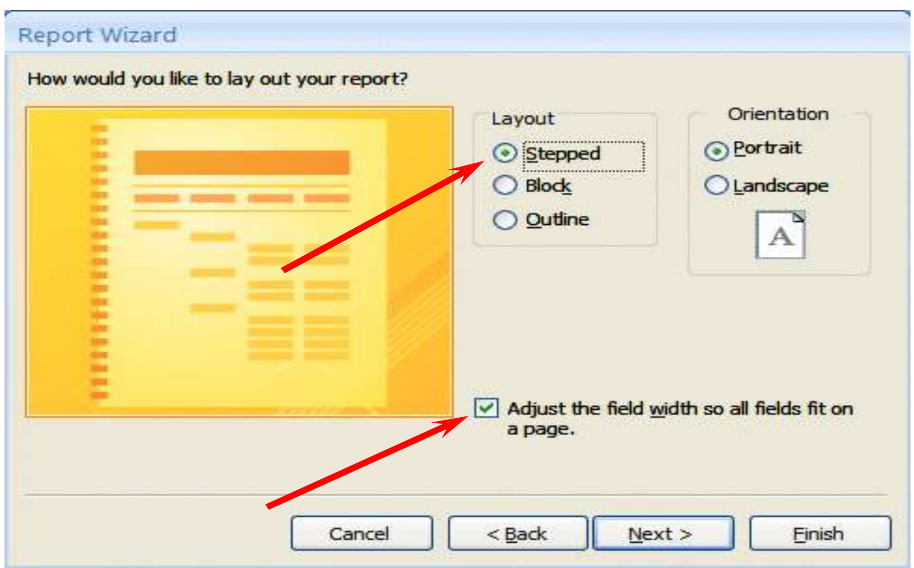
Name order. Notice the **Ascending** button to the right of the Sort boxes. This indicates that the Field that you select is in **A to Z** or **ascending order**. If you click on this button, it will reverse the order from **Z to A**, or **descending order**. Click on the small down arrow to the right of the first box and select **Last Name**. Leave the order as **Ascending**. Now, select **First Name** in the second box. When you are finished, your **Report Wizard** menu screen should look like the one **above**.

Notice a **Summary Options** button **below** the **sort fields** you have selected (Access 2007 is smart, isn't it?). Whenever you see the **Summary Options** box it is because Access 2007 knows that you selected a number field for your report. The Summary Options box ONLY appears when a number fields are selected. Click on the **Summary Options…** button.

The **Summary Options** menu box allows you to *enter calculations* for **numerical** and **currency** fields if you have selected any. It will *summarize* these calculations by each group and in **total**. So, since **Salary** is a **currency** field, we can obtain calculations. Click in the **boxes** under **Sum** and **Avg**, this will perform these calculations (*as you will see in the report*). If you want percentages as well, click in the box next to Calculate percent of total for sums.



Click on **OK**. This will return you to the previous Wizard screen. Click on **Next** > again.



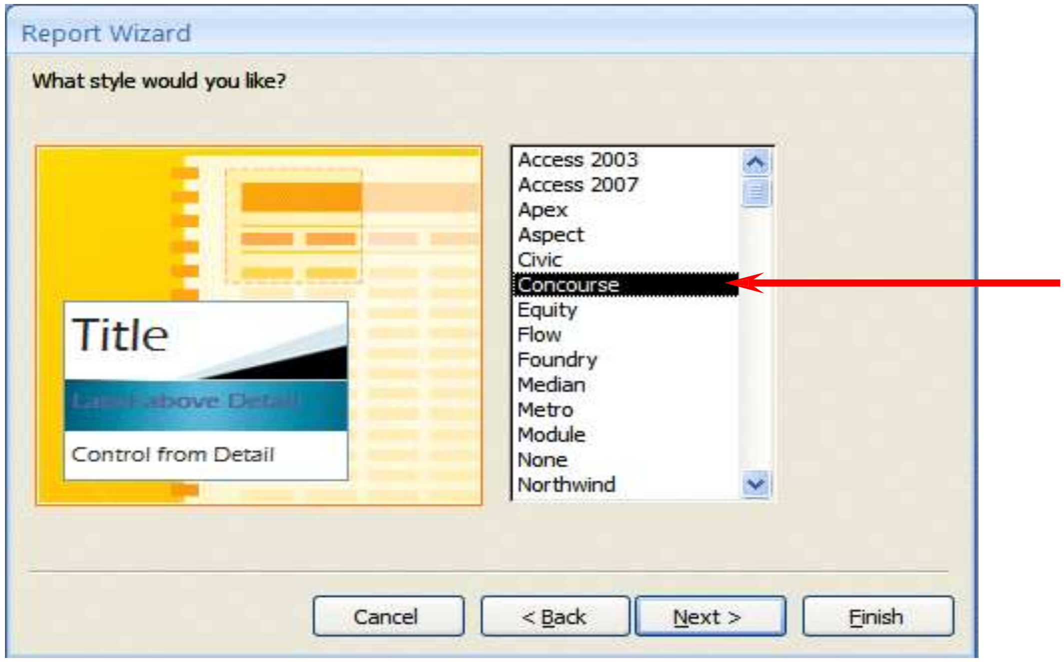
This Report Wizard screen allows you to select a **layout** for your **report**. **Click** in the small circles to the left of each choice in the **Layout area** and observe the results. For the moment, we'll stay with the default layout, **Stepped**. So click back on the circle corresponding to the **Stepped layout**. Leave the report in **Portrait Orientation.**

Note: At the bottom of the last **Report Wizard** menu screen (*see image above*) there is a check-in small box to the left of **Adjust the field width so all fields fit on a page**. *This is a very important check.* This means that no matter how many fields you place in your report, they will all fit on one page. With a few fields in the report, this is no big deal. However, if you have a lot of fields, they will be all scrunched up and you may notice that some Field Names and data for these fields are slightly cut-off.

As mentioned at the beginning of the Reports section of this tutorial, this is where an advanced course or manual are essential (recommended).

Click-on **Next >** again.

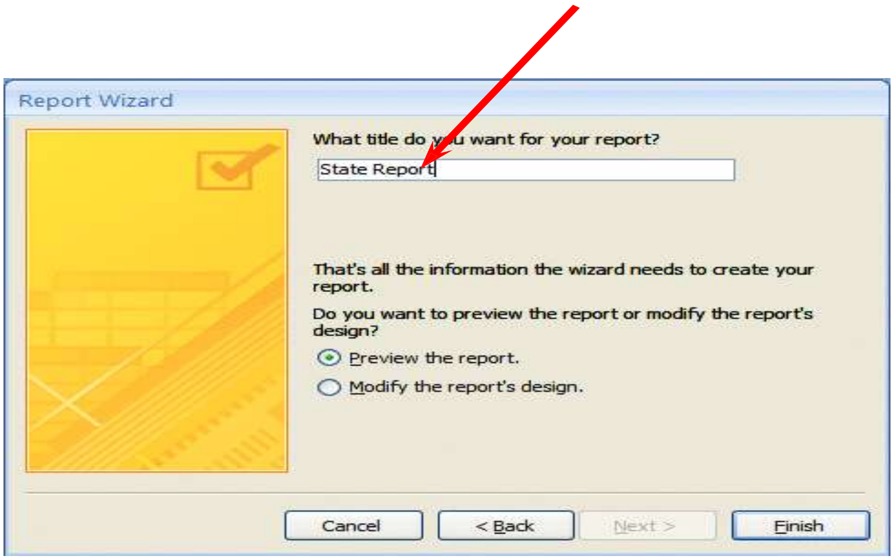
The next Report Wizard will appear.



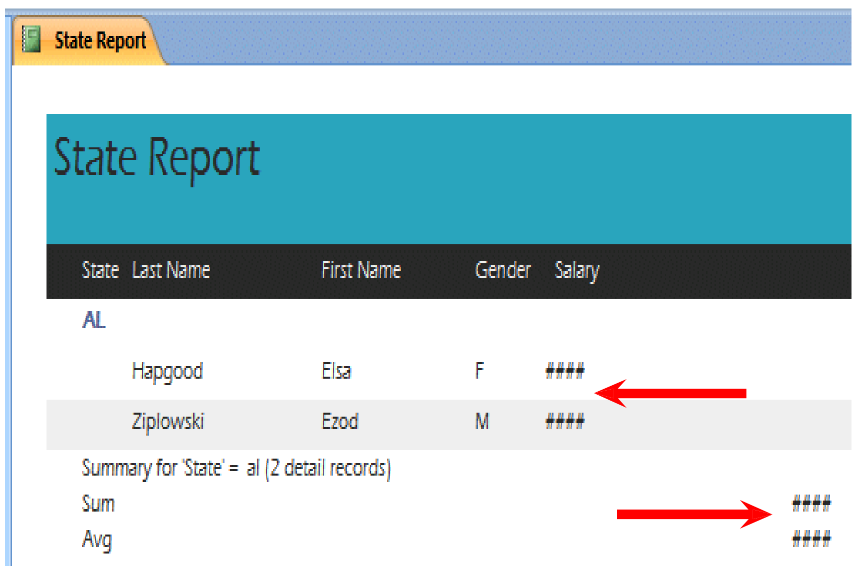
This menu screen allows you to select the **Style** that you would like for your report. Click on the **choices** (*Apex, Aspect, civic, concourse etc.)* and discover them. Choose which ever style you desire and click on **Next >**.

The next **Report Wizard** screen is the last screen in the sequence. It allows you to select a title different from the name of your database, incase you want to make some changes.

Note that the small circle beside **Preview the Report** is highlighted. When we click the **Finish** button, Access 2007 will display a **preview** of your report. We'll name this report **State Report**. Use this name, or any name you desire, and click on **Finish**.



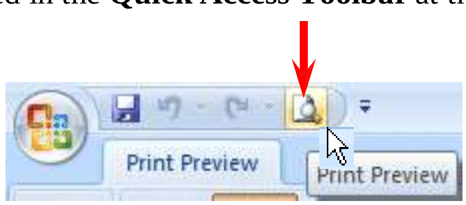
This is a report in **Tabular** (*Columnar*) format. Your screen should look like the one below.

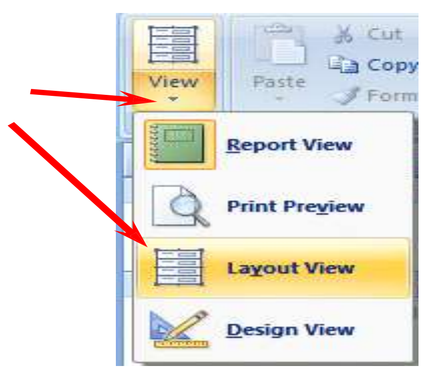


Notice the #### symbols under **Salary** and on the right of **Sum** and **Avg**. This scenario arises because the displayed fields are not wide enough to display these numbers. We'll show you how to adjust column widths in a moment.

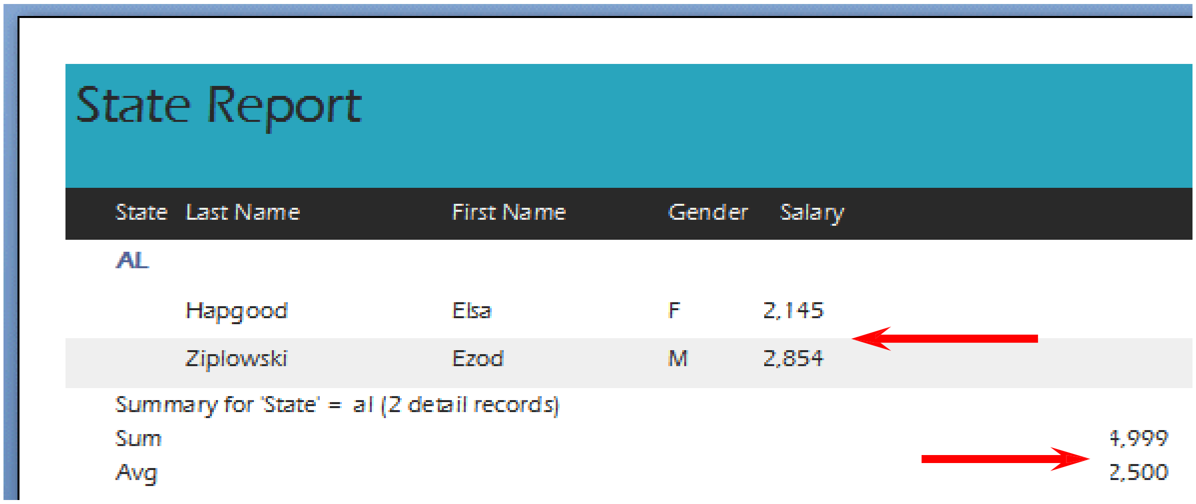
You will Notice in the upper left Corner of the Access screen that you are currently in **Layout View**. If you click the small down arrow under **View**, the drop down menu on the right will appear.

To really see what the **State Report** will look like, you'll need to go to a **Print Preview**. You can either click on the **Print Preview** selection in the menu on the right, or click the **Print Preview** button you placed in the **Quick Access Toolbar** at the top left of your screen.

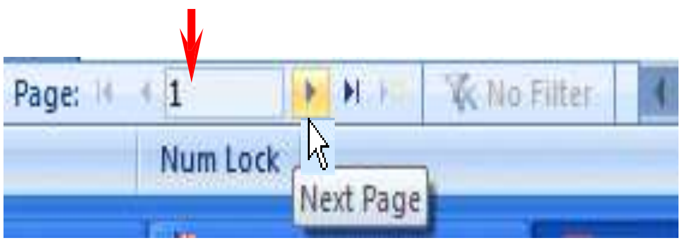




In the **Print Preview** image below you can see that the **Salary, Sum** and **Average amounts** are cut-off. We'll need to go to the **Design View** to enlarge these boxes so that we can see all the numbers.



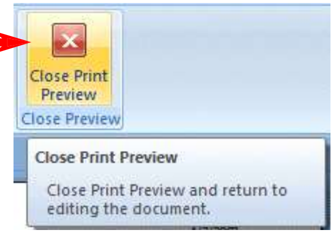
You will notice in the lower left corner of the State Report screen that you are on Page 1 of the report.



The triangle arrow buttons to the left and right of Page 1, will take you to the first page of the report, the previous page, the next page, and the last page. Try clicking on them.

You will also notice that your cursor in this **Preview Report** screen is a *magnifying glass*. This shows you how a page of your report will appear when you print it. Each time you click the magnifying glass you will *"zoom in" or "zoom out"* making your report appear larger or smaller. You will zoom to the section (*area*) where you place your magnifying glass just as if you were using a real magnifying glass on a real piece of paper. You'll magnify the section (*area*) where you are holding the magnifying glass. So, give it a try.

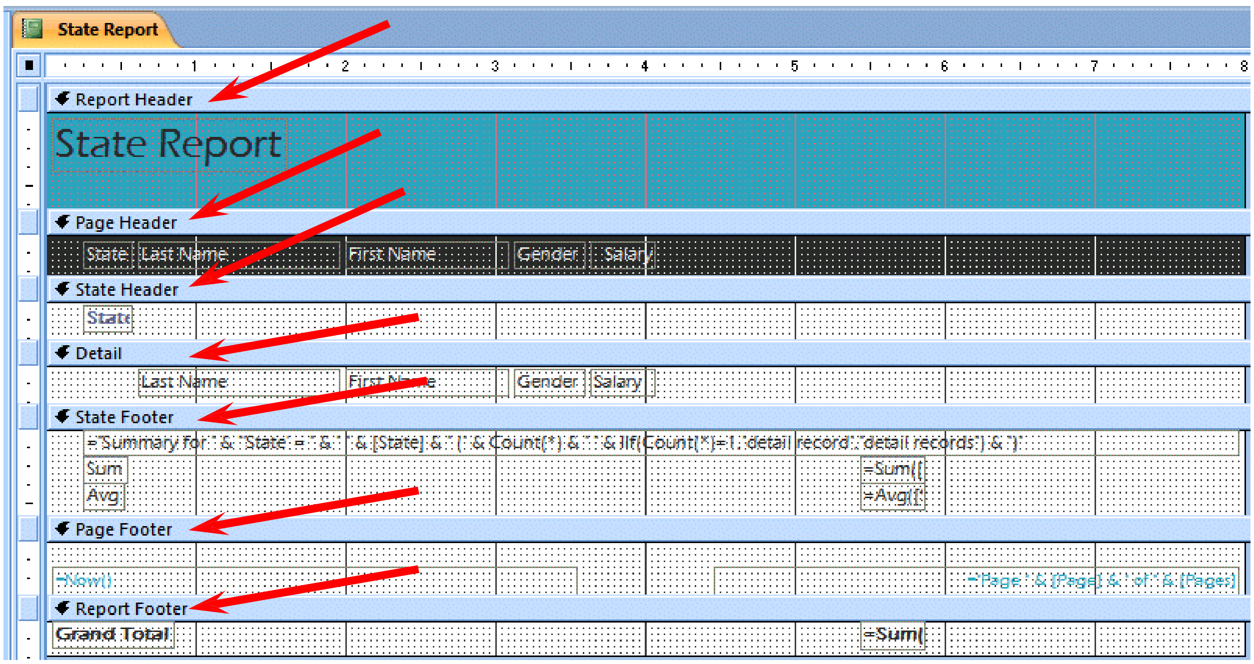
To return to your **State Report**, click on the **Close Print Preview** button on the right side of the **Print Preview Tab/Ribbon.**



This will take you back to the **Layout View** of your Report.

You will see your **State Report** with all of the **####** symbols.

Click the down arrow below the **View button** in the upper left corner of your Access screen and then click the **Design View** selection.



We'll explain about what you see in the image above on the next page.

1.9.2 New Report Design Tools Tab/Ribbon in Access 2007

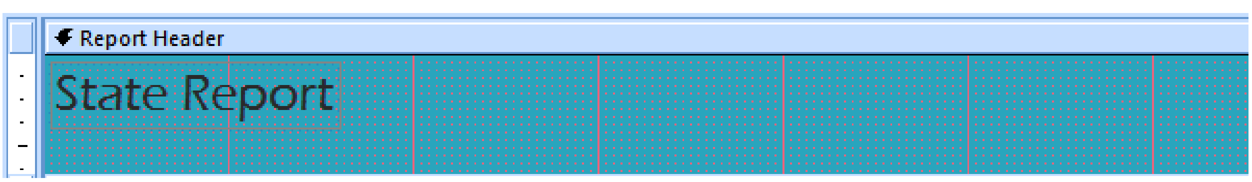
In addition to the change to the **Design View**, you will see that the Home Tab/Ribbon is replaced by a **Report Design** View Tab/Ribbon. In previous versions of Access, small menu windows would open for Design View. These have now been placed in the Tab/Ribbon you see below.

We enlarged a part of this Tab/Ribbon so you can see that the Tab/Ribbon Groups – Grouping & Totals, Gridlines and Controls have replaced the "*old*" menu boxes.

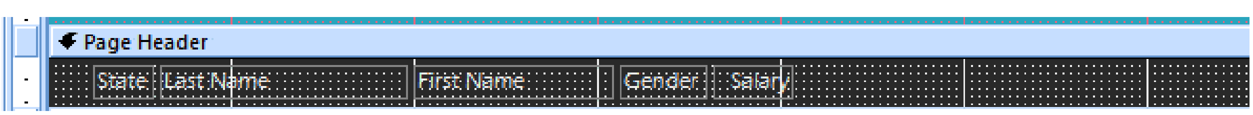
When you become more comfortable with Access Reports, you'll find that these tools come in quite handy.

Now, about the areas you saw in the **Design View** on the previous page, you would notice, to the left in the light blue part of the screen, as it indicates**: Report Header, Page Header, State Header, Detail, State Footer, Page Footer and Report Footer** (*see arrows on the previous page*)**.**

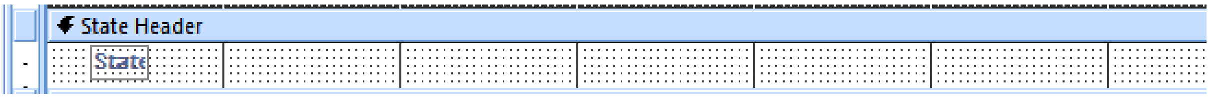
Report Header: If something appears here, it will only be shown on the first page of the report.



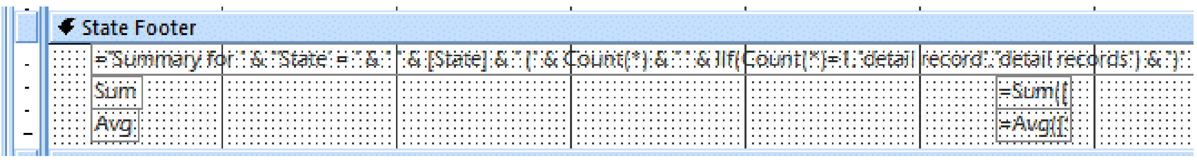
Page Header: If something appears here, it will be displayed on *each* page of the report at the top of each column.



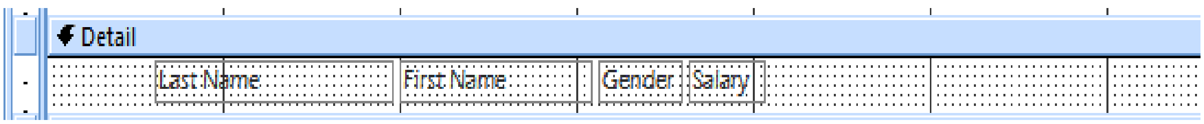
State Header This *sets-off* the State Grouping.



State Footer This *ends* the State Grouping.

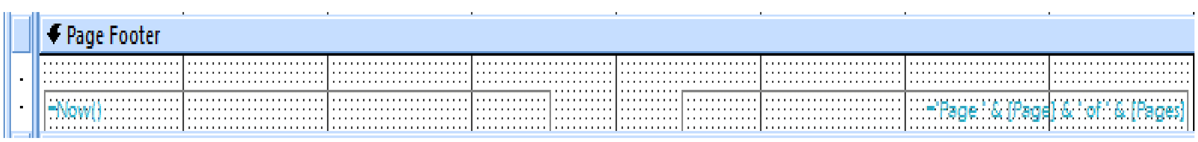


Detail: These are the field names from our database. Access will pull the data from the individual fields from our database records.



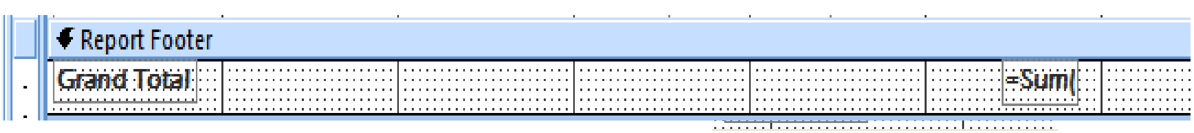
These are the database fields themselves. The fields print each time there is a person in the database. This field information is drawn from the database. *As you enter more people in the database and run the report again, more people will be displayed*. The "*size*" of the box you see on the screen was created when we created the field sizes.

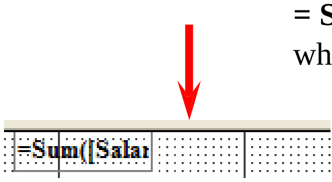
Page Footer: This is what shows at the bottom of each page.



Report Footer: This is information which is displayed ONLY on the last page of the report.

Also note, the lower right corners of the **State** and **Report Footer** area boxes indicate:

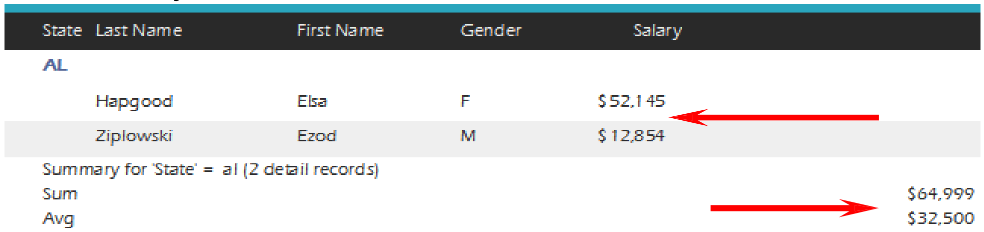




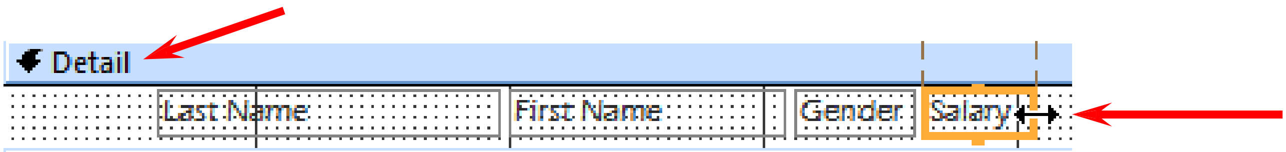
= SUM ([Salary]). This is a calculation box the Wizard created. This is what gave you the calculations for your **average** and the **sum** of the salaries in the **State** area and the **grand total** of all salaries in the Footer area.

Enlarging the Salary, Sum and Average "boxes"

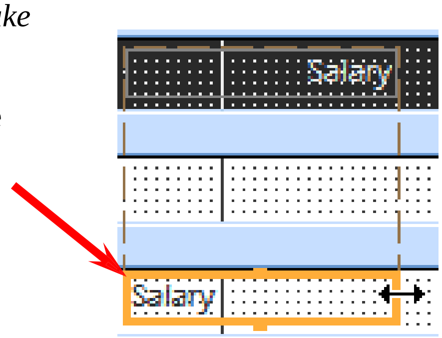
On Pages 43 and 44 we saw that the **Salary, Sum** and **Average** numbers were **cut-off**. We'll now show you how to widen the boxes so you can see all the numbers.



In **Design View**, click on the **Salary box** in the **Detail area**. The border around the box will turn *orange*. Carefully move your cursor over the right side of the box, you will see a *two-headed arrow.* Your **Detail** area should now look like the **image below**.

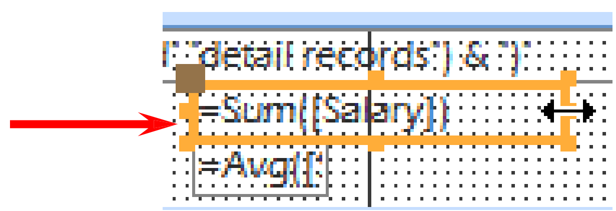


When you see the *two-headed arrow*, click and hold down the left mouse button and move your cursor slowly to the right. You will see the **Salary** box get larger, as well as the **Salary** area in the Report Header. *Take your finger off the left mouse button.*



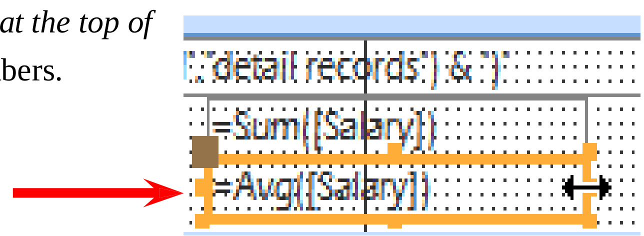
Your **Salary** box and **Salary Header** should now look like the image on the right. *If you have not made the area large enough, or too large, you can return to the Design View and adjust as need be.*

Now move to the **State Footer** area and do same for the **Sum** box**.**



And, then the **Average box**.

Click on the **Print Preview** button (*like you did at the top of Page 44*). You should now see all of your numbers.



1.9.3 Saving Reports

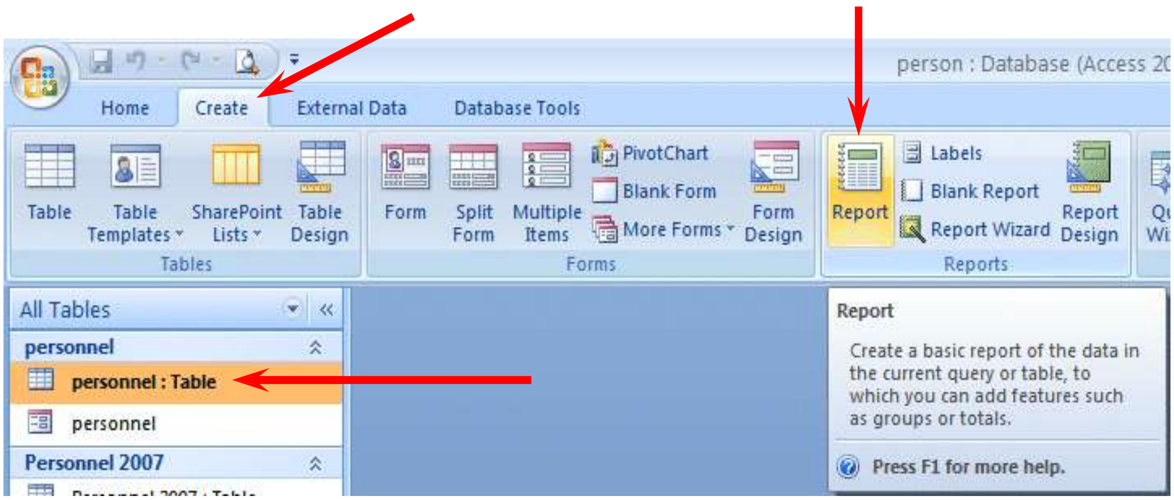
Since you have already named your Report (*State Report*), move your cursor over the **State Report** Tab and click the right mouse button. When the drop down menu appears, click **Save**. You will see your **State Report** in the area on the left of your screen with your Table, Form and Query.

Close this report again as you did previously (*by right clicking on the state report tab and selecting close*).

1.9.4 Quick Reports

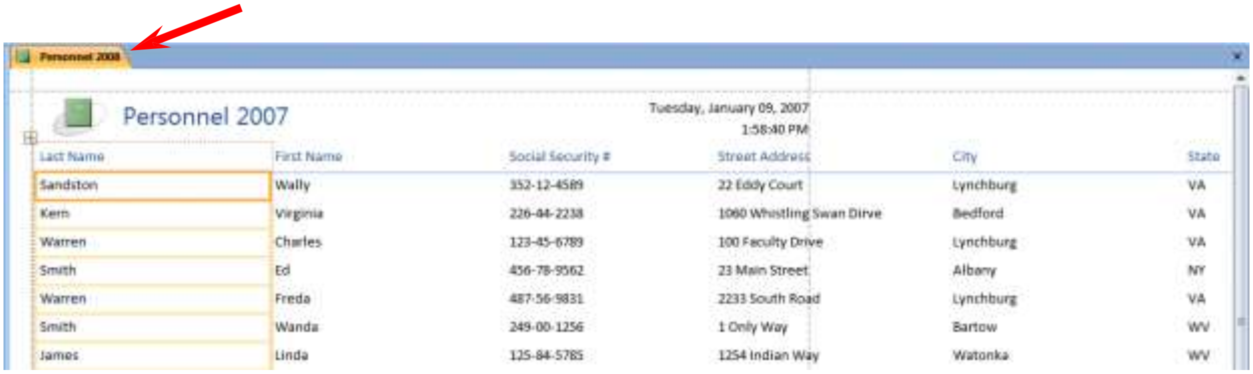
On Pages 36 and 37 we showed you how to create a report using a **Report Wizard**, this was just to give you a *feel* for how Reports are produced. A good and recent feature of Access 2007 is what we'll call **Quick Reports**. Now that you know how to use the **Wizard** and understand basic reports, you can create similar – simple reports – with a couple of clicks.

Click the **Create** Tab. Then, click your **Personnel selection** under All Tables. Move your cursor over the Report selection in the Reports group. Read the pop-up Help box and then click Report.



You will see several things occur at the same time. On the lower right side of your Access screen you will see the **Quick Report** (image at the top of the next page).

Notice that Access has chosen a name for the Report based on the name of your Table.



Also notice that a new **Report Layout Tools** Tab with a Formatting Tab/Ribbon appears to assist you.

You can carryout tests with this Report and the Formatting Tab/Ribbon as you desire. Don't worry if you really create a Report that you don't want. You can use the ***Undo*** *Arrow to go back a step or two or you can right click on the new Report Tab and select Delete.*

As you create reports you may save or not save, as you desire.

Reports can become complex, very quickly. This is only an introductory tutorial, which furnishes a simple guide to report design. You might want to purchase a book on Access or try a separate tutorial on reports. Our favorite book is ***"Microsoft Press Access 2007 – Inside Out"*** from Microsoft Press

When you are finished simply exit Access as you did on **Page 19**.

Now that you have the basics, you might want to try some things on your own. Try using the Wizards in Table, Query and Reports.

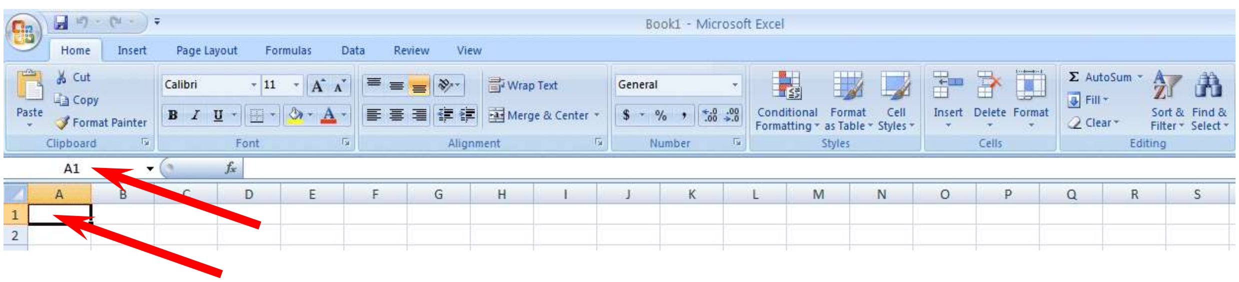
2. MICROSOFT EXCEL 2007

2.1 Starting Excel 2007

In the following exercises you will learn some of the necessary steps to **create** a **spreadsheet** using **Microsoft Excel 2007**. You will learn not only how to type various items into the spreadsheet, but also how to *copy columns, widen columns, fill columns, add, subtract, multiply, divide, do graphics, just to name a few.*

To begin, load the spreadsheet by quickly clicking twice on the **Excel 2007 Windows Icon** in the Windows Desktop. If you do not see an **Excel Icon**, click the **Start** Button in the lower left corner of the screen (*incase you are using Windows XP. Vista, 7 and 10*), move the cursor up to **Programs**, then move to **Microsoft Office.** Move down to **Microsoft Excel 2007** and select it.

A spreadsheet is a number manipulator. To make the handling of numbers easier, *all* spreadsheets are organized into *rows* and *columns*. Your initial spreadsheet will look like the one below:



Notice that the main part of the spreadsheet is composed of **Rows** *(Labeled 1, 2, 3, 4, etc.)* and **Columns** *(Labeled A, B, C, D, etc).* There are a lot of rows and columns in a spreadsheet. The *intersection* of each row and column is called a **cell**. In the image above the cursor is on the *home cell* – **A1**.

Notice that **Row 1 and Column A** are *highlighted* and colored *orange*. This indicates what is called the *address of the cell*.

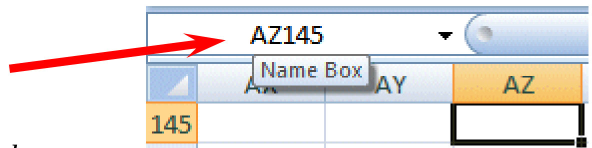
Slightly above **cell A1**, **A1** is displayed in a **small box** called the *Name Box*. Whenever you click on a cell the address of that cell will be shown in the Name Box.

If you have used previous versions of Microsoft Excel you will quickly notice that the above image is very different from what you are used to seeing. In Excel 2007 you will now use Tabs, Ribbons and Groups, as well as special Tabs/Ribbons. These replace the Menu Bar and Buttons in older versions.

In this tutorial, whenever we indicate that you need to click the mouse, it will mean to click the left mouse button – unless we indicate that you should click the right mouse button. So, always click left unless we tell you otherwise.

Moving Around the Spreadsheet

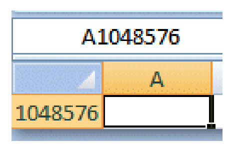
You can move around the spreadsheet/cells by clicking your mouse on various cells, or by using the *up*, *down*, *right* and *left* arrow movement keys on the keyboard. Or, you can move up and down by using the *elevator bars* on the right and bottom of the spreadsheet. Go ahead and move around the spreadsheet. Hold down the down arrow key on the keyboard for a few seconds – then click on a cell. Notice how the Name Box always tells you "*where you are"*.



Now hold down the **right arrow key** on the keyboard for a few seconds. Notice how the alphabet changes from single letters **(A, B, C,. …. Z)** to several letter combinations **(AA, AB, AC).** There are hundreds of columns and thousands of rows in a spreadsheet. Anytime you desire to return to the **Home Cell (A1)** simply **click in** the **Name Box** and type-in **A1**. Then tap the Enter key and you will go to cell **A1**. You can go to any cell by this method. Simply type-in a row and column, tap the Enter key and you'll go to that cell.

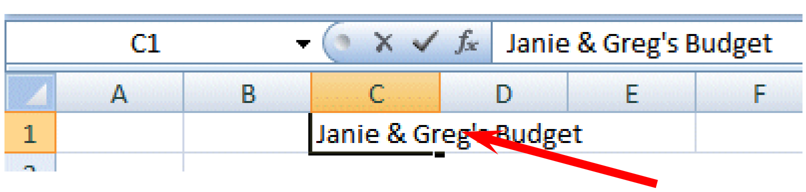
If you want to go to the last column on the right, hold down the **Ctrl** key and tap the right arrow key.

If you want to go to the last row at the bottom, hold down the **Ctrl** key and tap the down arrow key.



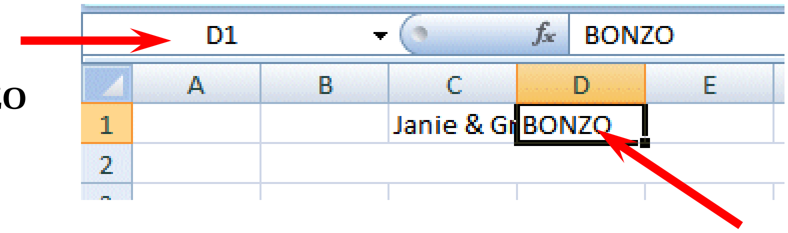
Now that you have the feel of how to move around the Excel spreadsheet, go to the cells as indicated below and type-in the following:

C1 **(Your Name)'s Budget.** It should look like the image below. Do not tap **Enter** when you finish

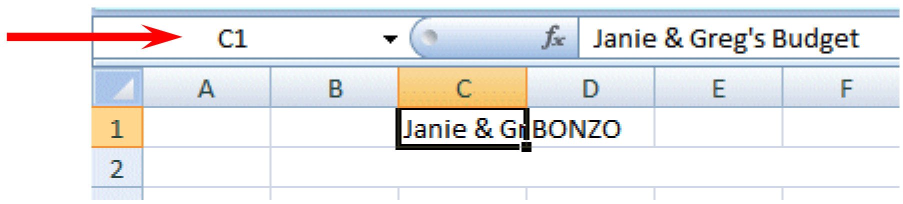


Look at cells **C1** and **D1**. Notice how your entry has spilled over from **C1** into **D1**. Sometimes this is a problem, and sometimes it is not.

Tap the Enter key and then click on cell **D1** and type-in the word **BONZO** and tap the Enter key**.**



Notice how **BONZO** now covers the right part of your original entry. Move your cursor over cell **C1** and click on it. Look at the upper part of the spreadsheet just above the cells where you typed BONZO. Your name and the word budget are still there! Bonzo only covered the portion in cell **D1**. See the image and arrows below**.**



There are several ways to take care of this. For the moment move back to cell **D1** and click on cell **D1**. Tap the **Delete** key (*above the arrow movement keys on the keyboard*). Notice that **Bonzo** disappears and your entire entry reappears. This is one way to expose the entry. We'll look at some others as we go along.

Now we'll continue entering text and data. We think that creating a simple personal budget would be a logical way to show you how a spreadsheet works. Move to the following cells and type-in the information indicated. You can click on each cell and then type-in the entries.

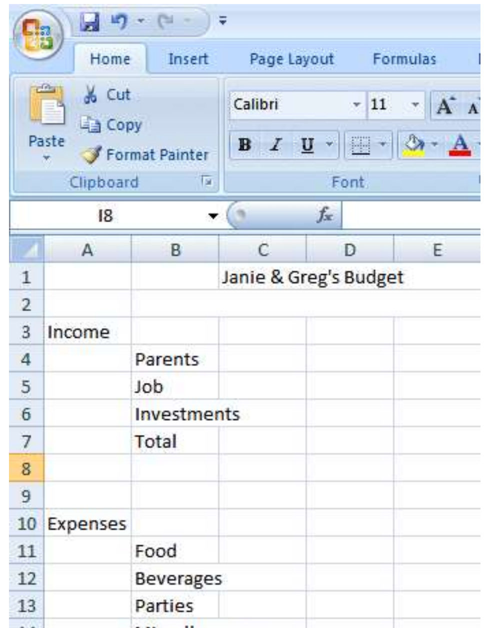
If you happen to make a mistake, simply retype the entries. Later on we'll see how to *Edit* mistakes. Any time you want to replace something in a cell you can simply retype the new entry and it will replace the old one.

Cell Type-in

A3	**Income**
B4	**Parents**
B5	**Job**
B6	**Investments**
B7	**Total**
A10	**Expenses**
B11	**Food**
B12	**Beverages**

B13 Parties
B14 Miscellaneous
B15 Total

Your spreadsheet should now be similar to the image on the right.



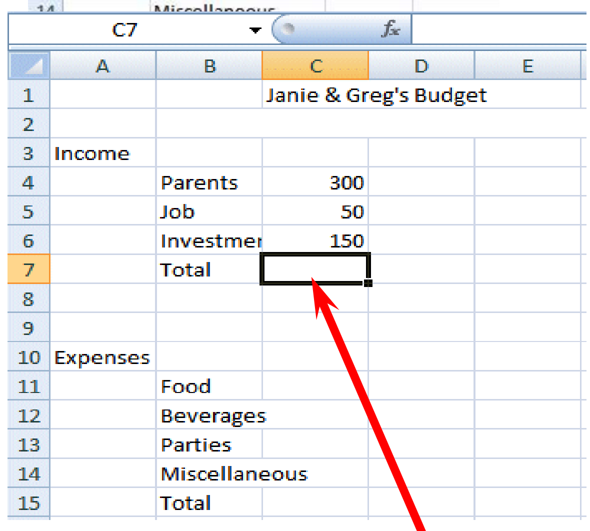
At this point you probably noticed, the words "**Investments**" and "**Miscellaneous**" run over the spaces given in the cells. Do not be concerned at this point. We'll soon fix this.

Now, **type** the **numbers** in the **cells indicated:**

C4 300
C5 50
C6 150

When you type-in the **150**, tap **Enter**.

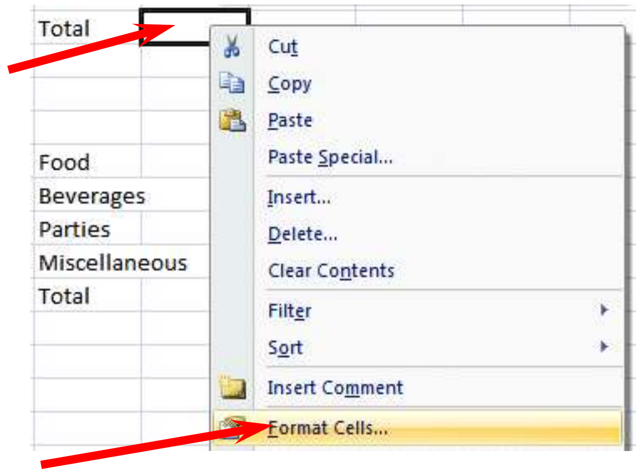
Your spreadsheet should look like the image on the right.



You will notice that when you enter text that the words line up on the left side of the cells. When you enter numbers, they line up on the right side. This is because we are using the United States (*English*) version of Excel. Other international versions will line up logically for their text and monetary forms.

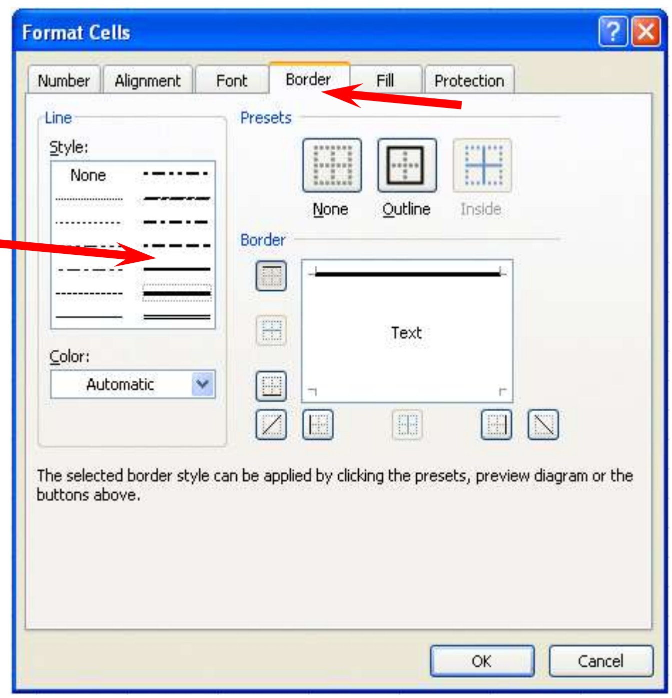
We would like to place an underline at the bottom of the three figures so that we can indicate a total below in cell **C7**. Point to cell **C7** (*with the mouse*). That's where we want the line --*always move the cursor to the place where you want to insert a line*. With the arrow on cell **C7** tap the right mouse button.

A sub-menu with a caption **Format Cells** appears.



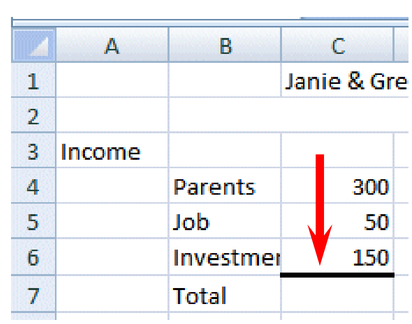
The right click will always bring up a menu that is tailored to the place where you click. This will work in any Microsoft Windows product. You can always tell where you click the right mouse button, for the cursor arrow will always be in a corner of the menu that appears exactly where you clicked the right mouse button.

Select **Format Cells.**



When the **Format Cells** menu screen (*below*) appears, select the **Border Tab**.

Look at the **Line Style** box on the right side of the menu screen. There are several types of lines that you can choose. Point to the thick **single line** in the **Style Area** (*see arrow on image beside*) and click the left mouse button. A box will go around the line. Look at the area which says **Border.** Point to the upper part of the Text box *(see arrow)* and click the left mouse button. A thick black line will appear at the top of the Text box.



If the thick line does not show-up at the top of the **Text box**, click again at the top line area in the Text box and the line will disappear. Then click on the thick, single line in the **Line Style** box again and repeat the previous instructions. Incase you make a mistake, simply click **on and off** in the Text line boxes. You will notice that the lines appear and disappear. This is called *a toggle* in computer language. So, work at this until you get the line on the top of the cell. We have indicated that we

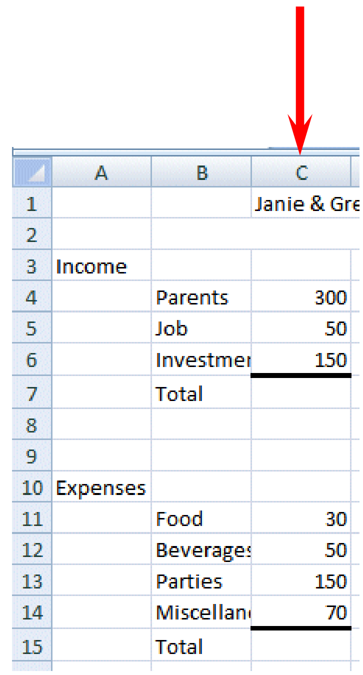
want a single thick underline at the top of the cell **C7**. Point to **OK** and click the left mouse button.

When you return to the spreadsheet, click somewhere other than cell **C7**. This is called *clicking away*. You should now see a line at the top of cell **C7**. Sometimes the box highlighting a cell hides the lines. If you messed-up, try again.

Now type in the numbers in the cells indicated.

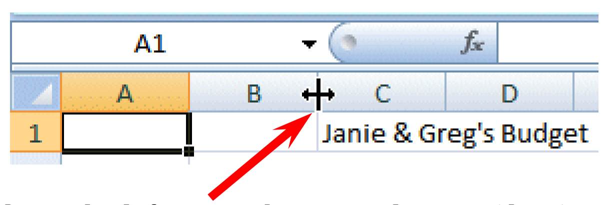
C11 **30**
C12 **50**
C13 **150**
C14 **70** **(**After you type 70, tap the **Enter key)**

Now, underline the top of cell **C15** as you did for cell **C7**.
Your spreadsheet should now look like the image on the right.



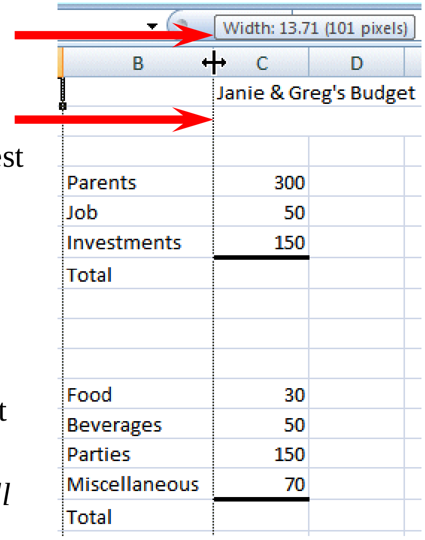
2.1.1 Widening Columns

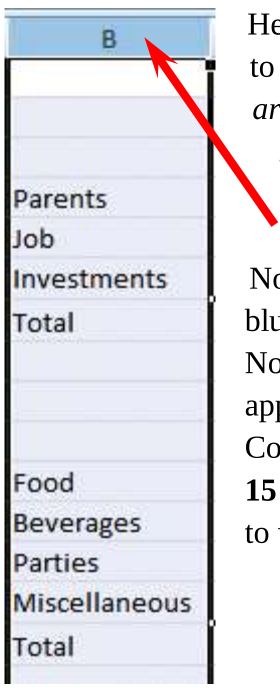
You probably noticed, as you typed in the numbers, some of the words were just too wide for the default **cell width** (*Investments and Miscellaneous*). Let's widen **column B** to take care of this.



Slowly move the mouse arrow to the right edge of the **B** cell (*between the B and the C*). The cursor will turn into an arrow pointing right and left with a small vertical line in the middle **(***see arrow below the column B on image beside***).** Holddown the left mouse button and move *(drag)* the line to the right.

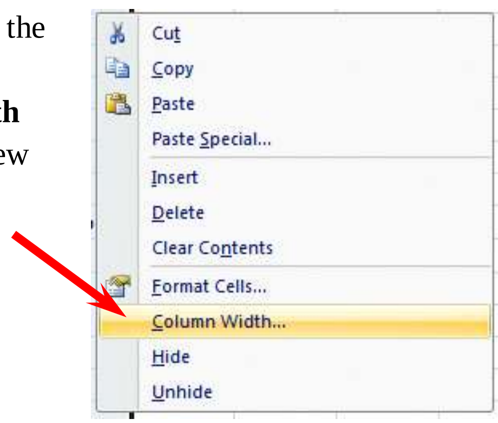
As soon as you start to move (drag) the mouse, a dotted vertical line will go down the spreadsheet and it will move as you hold down the left button and drag the mouse to the right. Keep moving your mouse to the right until you have included the widest word and a bit more (*for some space*). Release the button. The column is widened. Notice, above the two headed arrow cursor, that as you hold down and drag, it indicates the current width of the column.

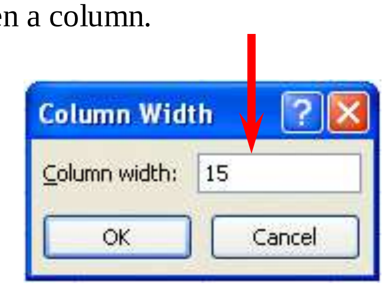




Here is another way to widen a column. Point to the **B** at the top of column **B** (*in the **Gray** area*) and click the left mouse button. *The cell should turn dark blue and the columnlightblue.*

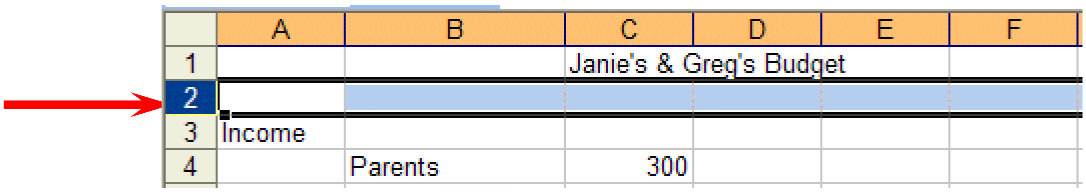
Now, keeping the cursor somewhere in the blue area, click the right mouse button. Notice that a **menu** with **Column Width** appears. Click on **Column Width,** a new Column Width menu appears. Type-in **15** and click **OK**. This is another way to widen a column.

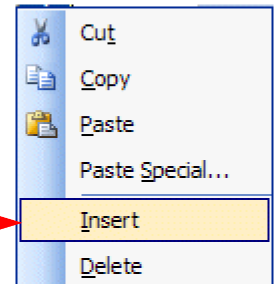




2.1.2 Inserting Rows

Oops... a mistake (*on purpose*). We haven't left enough room at the top of the spreadsheet to insert some budget months. So, move the cursor to the **gray 2** along the left edge (*this is the second row*) so we can insert two new rows. **Click** the left mouse button. You will notice that the whole row goes **light blue** and the **2** turns **dark blue.** Make sure the cursor arrow is either on the **2** or somewhere in the **blue row**.



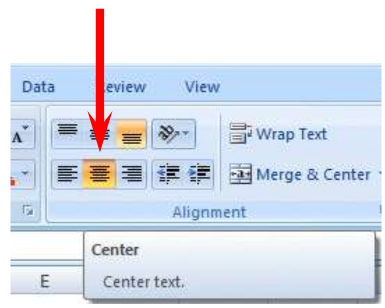


Click the right mouse button. A drop down menu will appear. Point to **Insert. Click** the left button on **Insert**. Notice how one row was inserted and how everything beneath the row, moved down. Do this again to insert another row. Excel and all spreadsheets, will remember where they moved your work and automatically adjust for these changes.
Income should now be in **cell A5**.

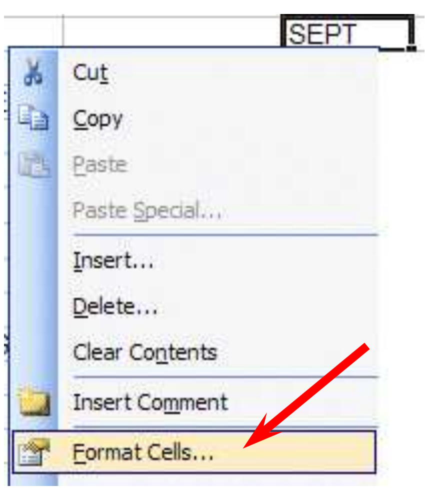
2.1.3 Aligning Cells

Now we'll type some more text. Go to cell

C3 **SEPT (**Type-in **SEPT** and tap the **Enter** key**)**

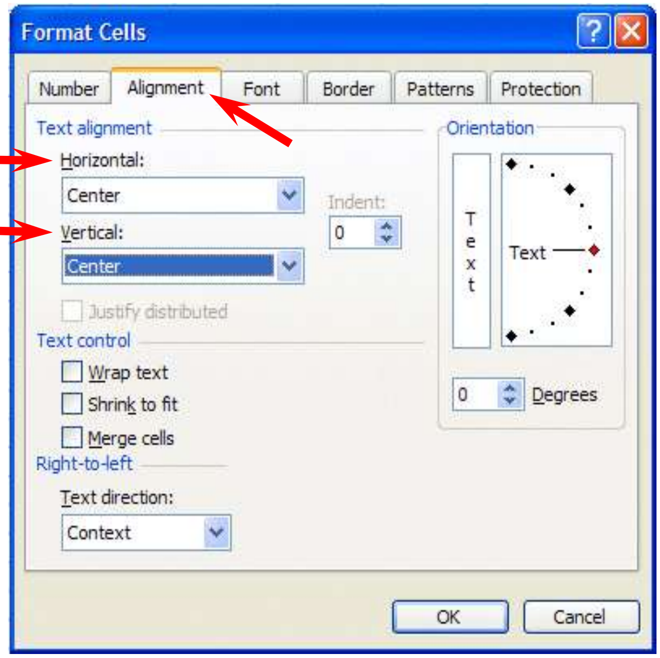


Notice how **SEPT** is automatically left aligned. Logically, since you are using Excel, the English version, the text is left aligned so that all of the text entries will line up nicely in the column cells. We would like to center **SEPT** in cell **C3**. Click on cell **C3** to mark the cell. One way to center **SEPT** is to simply click on the **Center** button in the button bar at the top of the screen. Make sure that you are on cell **C3,** then click on the **center** button (*see image above*). You'll notice that **SEPT** is now centered in cell C3.



Here is another way to center **SEPT**. Click right on cell **C3**. Then click on **Format Cells.**

When the **Format Cells Menu** appears, click on the **Alignment Tab** and then click on > **Horizontal** > **Center** > **Vertical – Center,** then click **OK**. Try it.

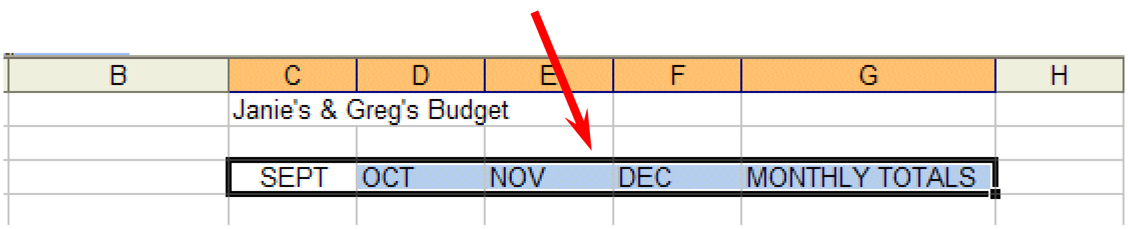


This is how you can align words for neatness. You can also point to several cells you want aligned and do this. We'll try that next.

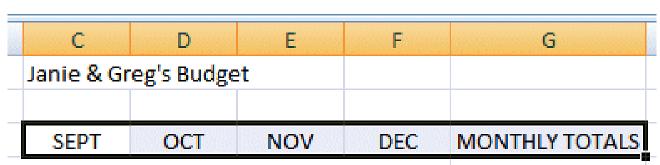
Now type the below text in the cells indicated.

D3 **OCT**
E3 **NOV**
F3 **DEC**
G3 **MONTHLY TOTALS** (tap the **Enter** key and then widen the width of Column **G**)

Next we'll highlight cells **C3** through **G3**. To do this, point to **C3** and click the Left mouse button. Then holding down the left mouse button, drag (move) the mouse to the right through **G3,** when the cells are highlighted take your finger off of the left mouse button.



Then point to the group of cells and click the right mouse button to bring up the **Format Cells** menu. Click the **Alignment Tab** and choose **Center** (*vertical & horizontal*). Then point to **OK** and click the left mouse button. All of the cells will be as centered. You could also click the Center button as you did before.



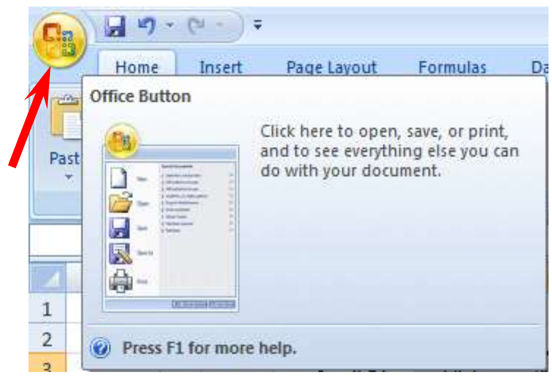
`

Don't forget to widen Column **G** and **MONTHLY TOTALS (***You know what to do***).** Move the cursor over the line between cells G and H and drag the line to the right to widen the G column, just like you did a few minutes ago.

2.1.4 Saving Spreadsheets

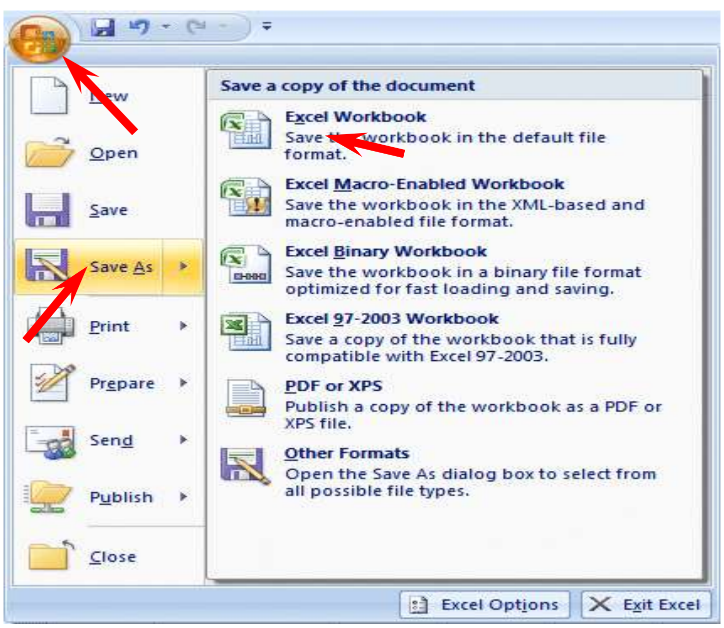
We have done quite a bit of work so now is a good time to save your spreadsheet.
If you have used previous versions of Microsoft Office, 2007 Office will be quite different in many ways. You've already noticed the **Tabs** and **Ribbons**, and that there is **no File choice** in a **Menu Bar**. Many selections have changed significantly in 2007 Office. This is one of them.

2.1.5 Microsoft Office Button



The **Microsoft Office Button** has replaced **File** in the Menu Bar. In the upper left corner of your Excel 2007 screen you will see a button similar to the image on the right. This is the **Microsoft Office Button**.

Click the **Microsoft Office Button**.



You will now see the Excel 2007 Microsoft Office Button selections.

First, notice that many of the previous File Menu Bar choices are included in this menu (*they are all here, we'll show you.*)

When we move our cursor over **Save As** an expanded menu of Save choices appear on the right.

Notice that you can save your spreadsheet in many different formats.

If you save as **Excel Workbook**, it will save your spreadsheet in an ***.xlsx* format**. This will save your spreadsheet in an ***Extensible Markup Language (XLS) format***. This format requires less storage space and makes the spreadsheet more shareable with others. However, folks using

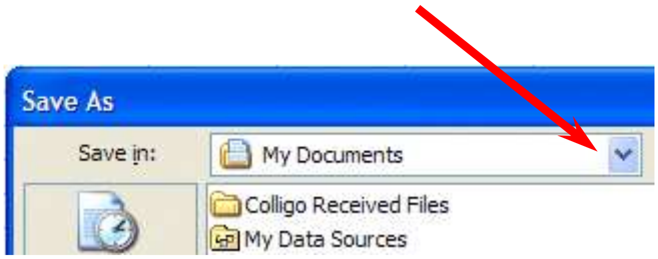
previous version may have a problem opening your spreadsheet (*and may have to download a special program to assist them*).

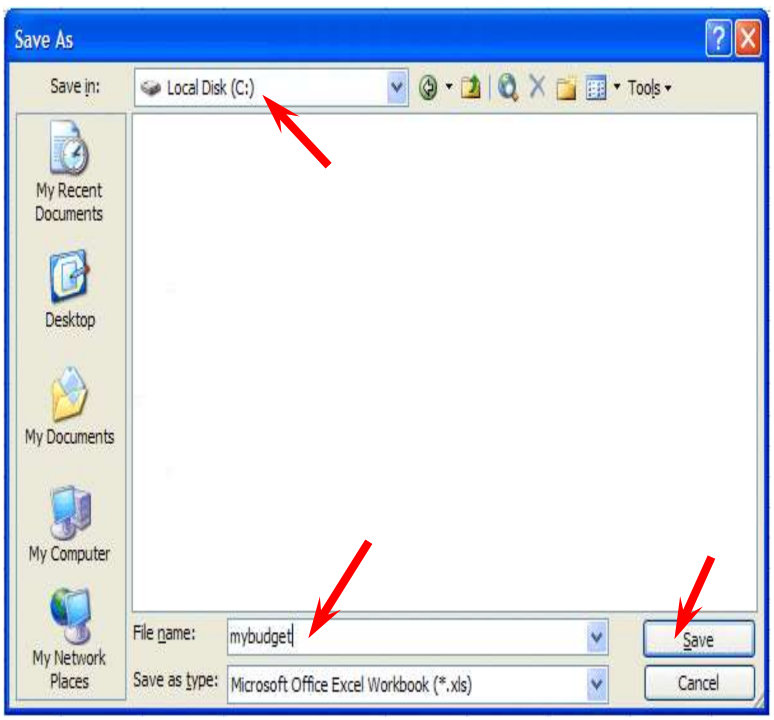
Many folks really like to save their files in *Portable Document Format (.pdf).* One of the neat new features of 2007 Office is the ability to save applications as PDF.

For this introductory Excel tutorial, we'd suggest that you save in the Excel 97-2003 Workbook format.

It's your choice, so you select the format you desire.

Notice in the upper left corner that there is a box to the right of **Save In** with a down pointing arrow to the right. Click on the arrow. This will show you all of the *drives* and *folders* where you may save your work**.**



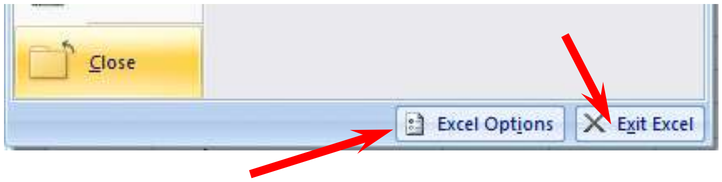


When you see the drop down list in the **Save in** area, choose the drive where you want to save your file. If you are going to use a diskette, put a formatted 3 ½ diskette in the **A Drive,** then click on the **3 ½ Floppy (A).** We are going to save our file on the **Local Disk (C:)**, our hard drive, so we chose that drive in the image above (*see top arrow*). To the right of **File name,** delete the information (*which is in the box*) and type-in **mybudget** (*see lower left arrow in image above*). This is the name under which you are saving your file. (In the future you will choose logical names for your spreadsheets as you save them.) Now point to Save and click the left mouse button (*see lower right arrow above*).

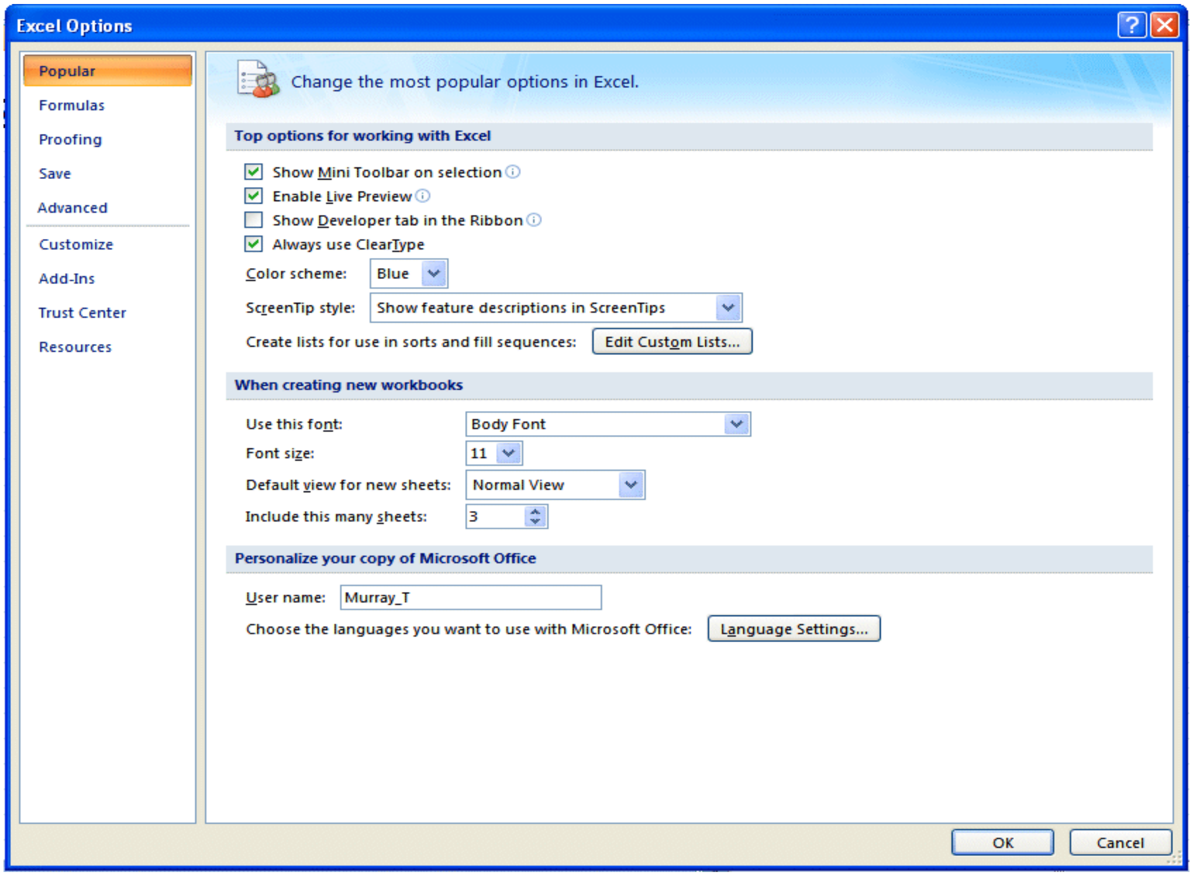
2.1.6 Exiting Spreadsheets

Anytime you need to leave your spreadsheet, click the **Microsoft Office** Button in the upper left corner of your

Excel screen and then click Exit Excel. If you have not saved your spreadsheet, a reminder box will appear asking you to do so.



Notice the **Excel Options** button to the left of Exit Excel. Earlier, we indicated that all of the choices under File in the Menu Bar are still available using the Microsoft Office Button. Click the **Excel Options button.** The **Excel Options menu screen (***below***)** will appear. As you can see, all of the choices available under File in the menu bar are here as well as many more.

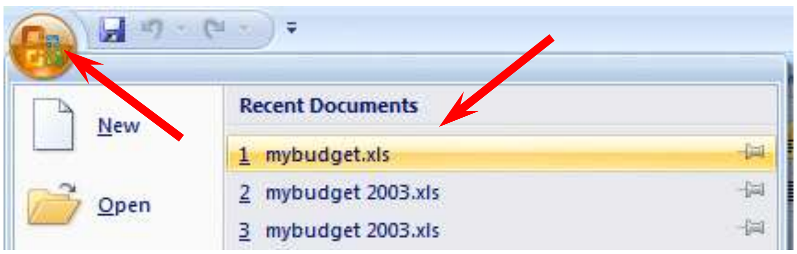


If you **click** the **Resources** selection in the Excel Options menu, you will see some great on-line resources available to assist you with Excel.

2.1.7 Retrieving Spreadsheets

When you need to return to a spreadsheet, **open** Excel, as you did on **Page 1**. When Excel opens, click the **Microsoft Office** Button in the upper left corner of the Excel screen.

When you click the Microsoft Office Button you will see, on the right of the Microsoft Office Button menu screen your spreadsheets (*Recent Documents*). Your **mybudget** should be on the list. Click on **mybudget** and your spreadsheet will open.



If you do not see your spreadsheet, click the **Open** button and follow the steps you used to save your spreadsheet (*on Pages 9-11*) except choose **Open**.

2.2 Adding Numbers

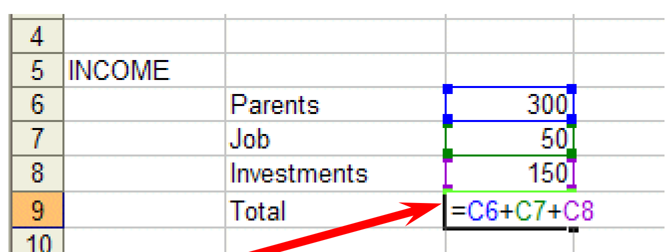
Next we want to learn how to add numbers. There are several ways to do this. Each method has its advantages and disadvantages.

Begin by moving your cursor to cell **C9,** and clicking on cell C9.
Always move to the cell where you want the answer to be located.

2.2.1 Type-In Method

We want to **add t**he three numbers in cells **C6, C7** and **C8**. To use this method **type-in** (*using the keys on the keyboard)* the following formula in cell **C9**:

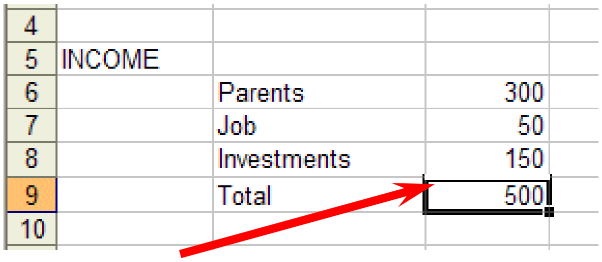
= C6 + C7 + C8



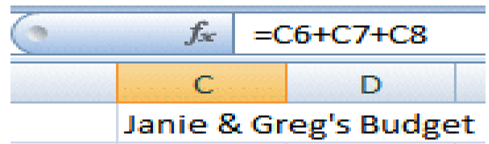
Your spreadsheet should look like the image to the right as you are typing-in the equation.

Note: *you don't have to use capital (upper case) letters – we only did this because they are easier to "see" in the tutorial.*

Now – **tap** the **Enter**key. Then, **click** on cell **C9again**. The **total** of these cells will now **appear** in **C9**.



When you have completed typing your equation, you will see this formula in the area below the menu bar.



Change the number in cell **C6** to **500** (*and tap Enter*). *See how the total automatically recalculates!!!*

This is one of the features behind the beuty of an electronic speadsheet!!

Whenever a number is entered in a cell the entire spreadsheet will automatically recalculate.

Something happened here!
Notice: you typed an equality sign (=) before the cell location. If you had typed in C6 + C7 + C8, Excel would have thought this entry was a *word* (*text)* and this entry would have shown as you typed it.

Try simulate this error and see the output. Any time you create an error in Excel, you can simply re-type or edit the formula to correct the error.

The **Type-in Method** is really easy if you have a *few numbers and can see their cell locations on the screen*. If you have a lot of cells in the formula, which are on several screens, this won't be a recommended method. The next method will work a lot better for numbers all over the place.

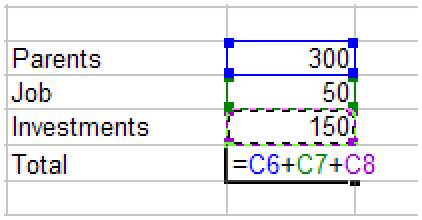
2.3 Subtraction, Multiplication, and Division

You can type a *minus* signe (**-**) for *subtraction*, *asterisk* (*****) for *multiplication* and *forward slash* (**/**) for *division*. As you become more verse with using these basic operators, we'll build some effective formulas using these features.

2.3.1 Point Method

Move to cell **C9** again and click on it. We'll now add the numbers a second way. Tap the **Delete key** on the keyboard to **delete** the **current formula**.

First, tap the equality sign (**=**) and then *point (move)* the cursor over cell **C6** and tap the left mouse button on cell **C6** (*you will see a marquee box go around the cell*). Now tap a plus sign (**+**) and move cursor to **C7**, tap the left mouse button, and tap another plus sign (**+**) and move the cursor to **C8** and tap the left mouse button (*notice how as you plus sign (+) and point, the addition formula is being built in cell* ***C9***), now tap **Enter**. The same formula can be built using the arrow movement keys on the keyboard (*except that you don't have to click each cell as the cell is marked when you move with the arrow keys*).



You would notice that, as you are entering the cell addresses while placing another (**+**) in the formula that the cursor returns to cell **C9**. Also notice, as you point to each cell that it is highlighted by a *marquee box,* this tells you what cell you've pointed to. *Pretty Good!*

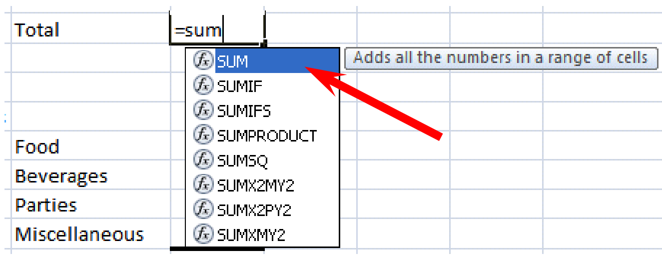
This method is good when you need to move to numbers that are spread out all over the place. Some people prefer this method to others, so it all depends on you the user and on the work load you have.

2.3.2 Function Method

Move again to cell **C9** and Delete the formula by tapping the **Delete** key.

Now **type-in** the following:
=**SUM(**

This tells Excel that we are going to sum some numbers in a RANGE which will follow the =SUM(



Notice another new feature found in Excel 2007**:** As you are typing **SUM** in cell **C9** that a **pop-up** menu appears under the cell. What you see are mathematical functions. One of these is **SUM**. As you become more comfortable with numbers in Excel, you can select the functions you need without typing in the entire function. We'll get into this a later.
There are *two* ways to put in this range:

2.3.3 Arrow Key and Anchor Method: With the keyboard arrow keys, move the cursor to cell C6. As you move, you will notice that the cell where the cursor is located appears after the =SUM(. When you get to C6 tap the *Colon(:)* Key. This is called an *anchor* and holds one end of the range in place. You will notice that a C6:C6 appears in the formula area under the button bar. This is a *one cell range*. Now move, with the arrow keys, to cell C8. See how cells C6, C7 and C8 are highlighted. This indicates the Range is **C6:C8**. Excel assumes, logically, that these

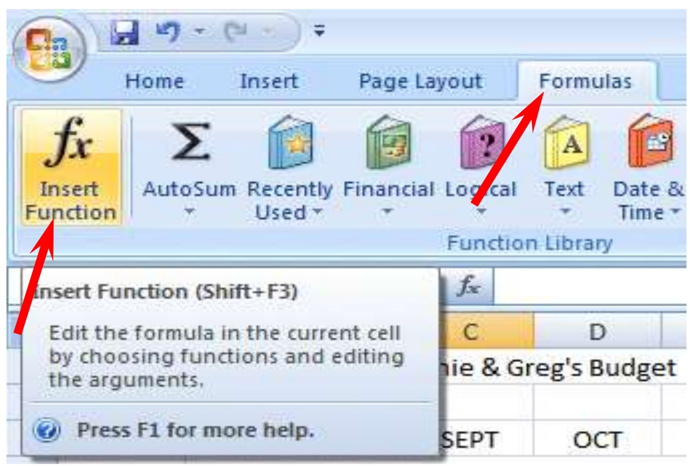
are the numbers you want to add. Now tap Enter. The numbers still add, but now the formula reads **=SUM(C6:C8)** instead of **=C6+C7+C8** like it did before.

2.3.4 Mouse Method: Move again to cell C9. Delete the formula in cell C9 by tapping the Delete key. Type in **=SUM(** as you did before. Point to Cell C6 with your mouse cursor. Click and hold down the left mouse button and move/drag the cursor down to Cell C8 (*Cells C6, C7 and C8 should be highlighted*) take your finger off the left mouse button. Tap Enter.

This **=SUM** Function is a *great way to add a lot of numbers, or a block/range of numbers.* By simply anchoring, and using page downs, or using the mouse, you can highlight lots and lots of numbers to add quickly. However, since it only sums you can't do subtraction, etc.

Point to cell **C9** again. Tap the **Delete** key to remove the formula currently in cell C9. This is an important delete, since what we'll explain below won't work correctly if you do not delete the formula in cell C9.

2.4 Functions

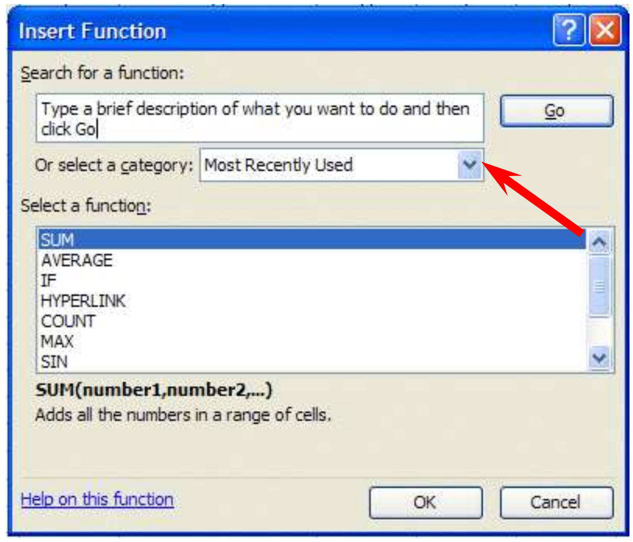


There are a number of formula is built into Excel, like **Sum**. These formulas are called **Functions**.

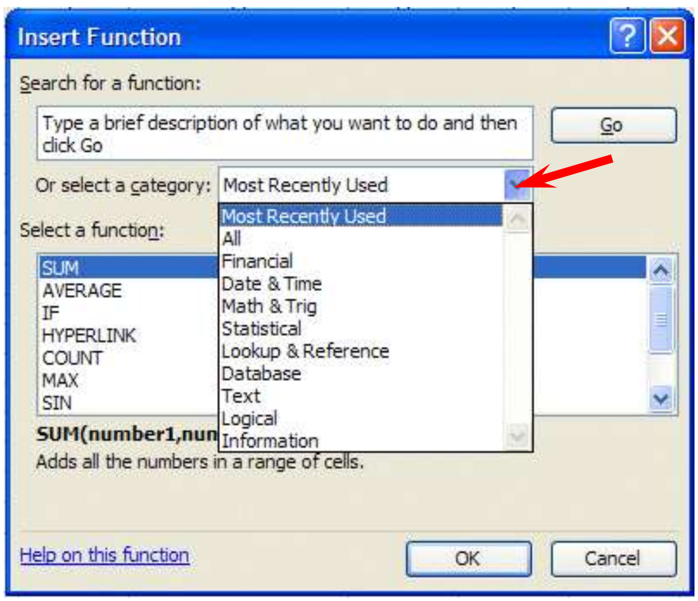
Another new feature found in Excel 2007 – Tabs/Ribbons. Look at the **top** of your **Excel screen** and click on the **Formulas** Tab. The Formulas Ribbon will display. On the left of the **Formulas Tab/Ribbon** is an I**nsert Function button.Click** the **Insert Function button**.

The **Insert Function** menu screen will appear (*image right*).

Let's work with the **Insert Function** menu screen**.** Click the small down arrow to the right of **Or select a category** (*see arrow at left*).

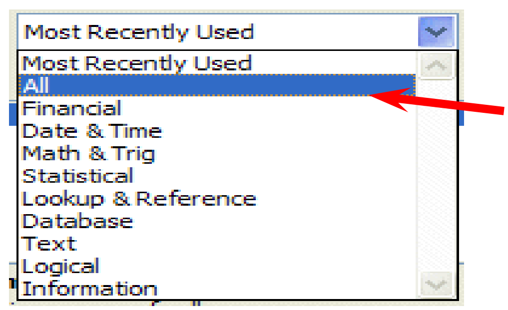


In the drop down menu that appears you can see that there are all kinds of formulas (*functions*) that come with Excel spreadsheet (*e.g. statistical, mathematical, financial, etc.*). Instead of having to go to math, financial, or statistical tables in a book, you can enter data from your spreadsheet into the formulas and receive answers.

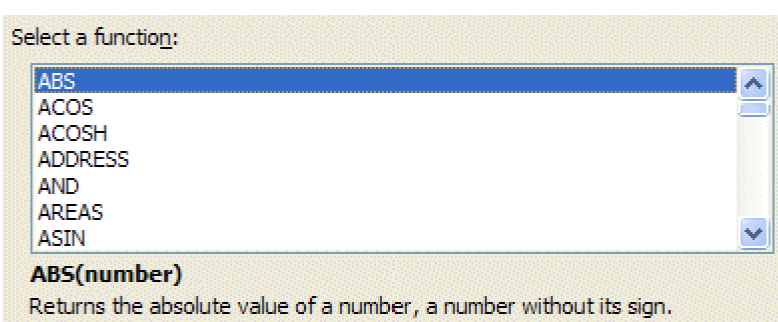


This is an really amasing and time saving feature. We'll now show you how to use the **Help features** of Excel 2007 to work with, and understand, these functions.

Click **All** in the **drop down menu**.



The Select a function menu will look like the image below.

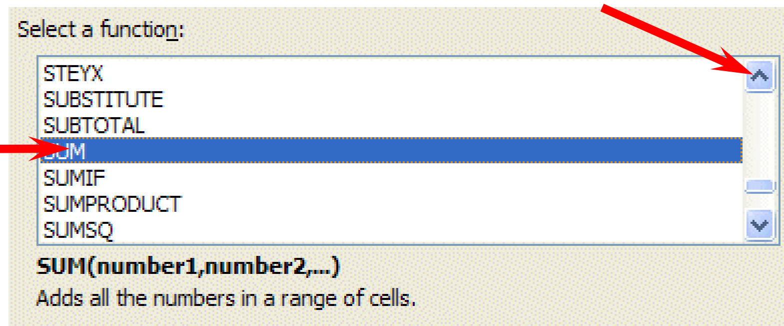


Look at all the functions (*formulas*). We'll just go through how to use the addition formula (*SUM*) in this tutorial. If you need these formulas in the future, you'll know they're here.

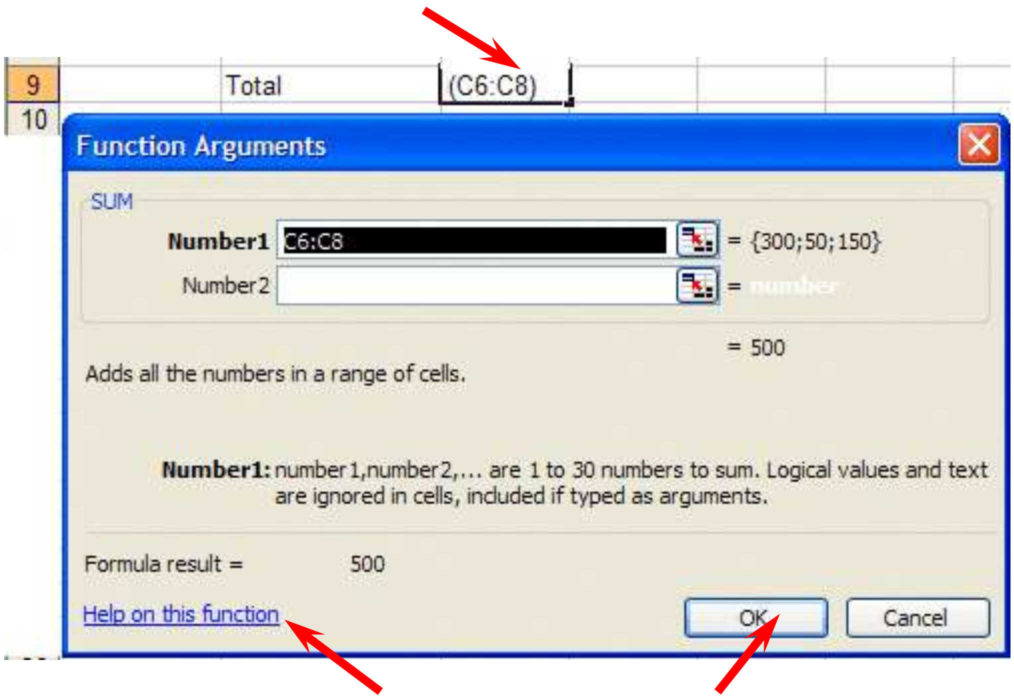
`

Use the elevator bar on the right side of the Select a function menu screen to move down the list until you see **SUM**.
Click **SUM**.

Then click **OK**.



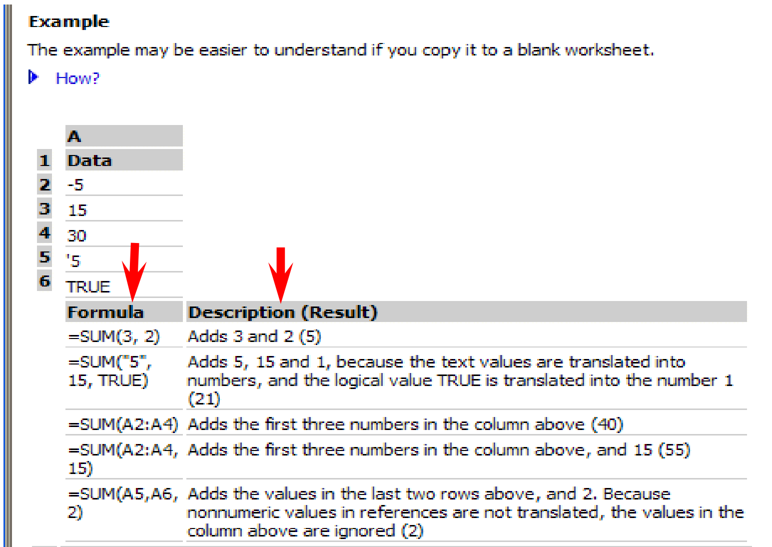
Remember, you clicked on Cell **C9** which was empty because you deleted the formula in that cell.



When you clicked **OK**, the **Function Arguments** menu screen (*above*) **appeared**. If you **look** at the top of the screen in the **SUM** area, you'll see that Excel 2007 has guessed that you desire to add the numbers above cell **C9**, where you clicked in your spreadsheet.
Notice that it indicates that **cells C6:C8** will be added (sum cells C6 through C8 – the colon (:) means *through*. It also indicates the numbers in cells C6, C7 and C8 and gives you the sum {300;50;150} = 500 (*right arrow above*).

But it's a little unclear how Excel did this. The **Help on this Excel Function** is excellent. So, to see how this SUM equation works, we'll go to Help. To do this, click **Help on this function** in the lower left corner of the screen (*see lower left arrow above*).

You will see a **Microsoft Office Excel Help window** appear (*similar to the one beside*) that will show you how to use this **SUM function** (*or any function*).



Example

The example may be easier to understand if you copy it to a blank worksheet.

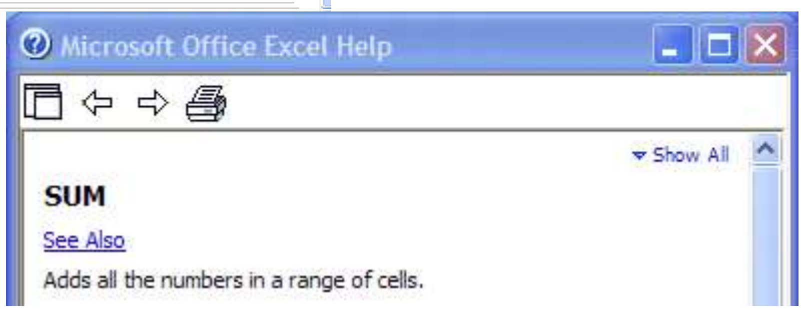
How?

	A
1	**Data**
2	-5
3	15
4	30
5	'5
6	TRUE

Formula	Description (Result)
=SUM(3, 2)	Adds 3 and 2 (5)
=SUM("5", 15, TRUE)	Adds 5, 15 and 1, because the text values are translated into numbers, and the logical value TRUE is translated into the number 1 (21)
=SUM(A2:A4)	Adds the first three numbers in the column above (40)
=SUM(A2:A4, 15)	Adds the first three numbers in the column above, and 15 (55)
=SUM(A5,A6, 2)	Adds the values in the last two rows above, and 2. Because nonnumeric values in references are not translated, the values in the column above are ignored (2)

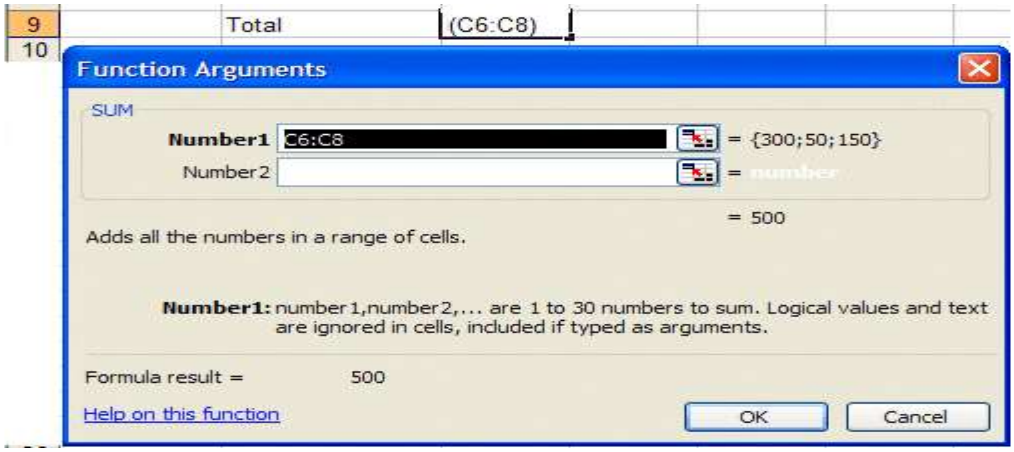
One of the really **neat** things about these Help windows is that there are examples for each function. We moved down the **SUM** help screen using the elevator bar on the right of the help screen. The bottom of the screen looks like the image below. Spend a few minutes looking at the SUM Help window and notice all of the features.

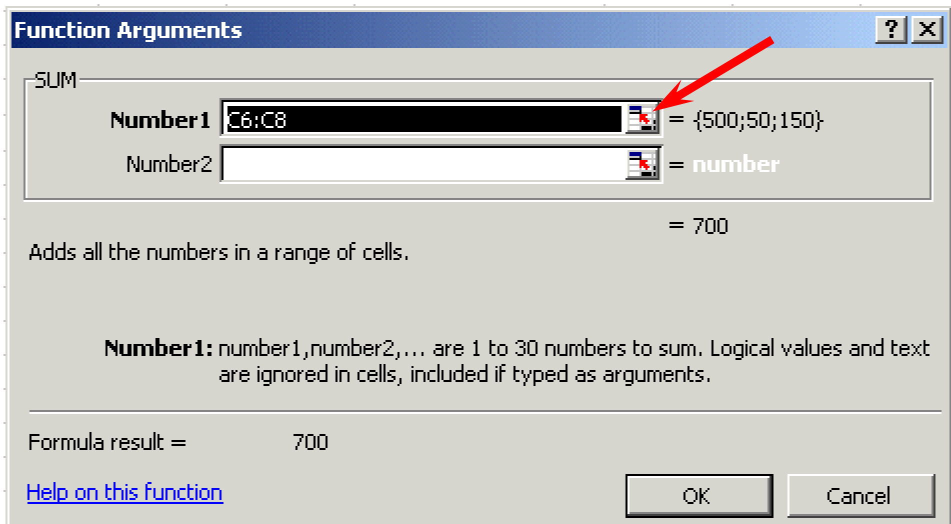
The bottom of the **SUM help screen** looks like the image on the left. Notice that it gives you examples from a small spreadsheet that has data in cells **A1** through **A6**. It uses these numbers in the examples at the bottom of the help screen.



When you have reviewed all of the help you care to see, carefully click the **X** at the upper right corner of the **Microsoft Office Excel Help** blue bar to close the Microsoft Excel **Help window.** *If you accidentally close the spreadsheet, simply reply yes to Save, and then re-open the spreadsheet as you did on Page 13.*

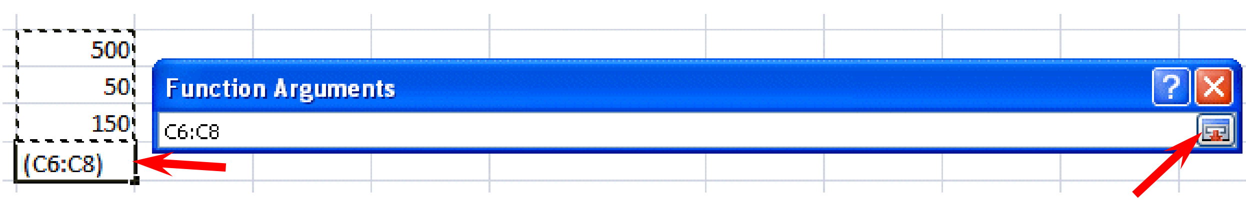
The **Function Arguments** menu screen will still be on the screen.





As you can see, in the area to the right of **Number 1**, the Wizard has guessed that you want to add the numbers in the range **C6 to C8, (C6:C8)**. Now that you are becoming skilled with Excel, we'll try something special. Carefully, point to some plain part, in the gray area above. Click and hold down the left mouse button, and **drag** the above **SUM** box away so that you can see your numbers in **C** column cells. When you have done this, release the mouse button. Now click on the small box on the right edge of the **Number 1 area** (*see arrow above*). It has a little **red arrow** in it.

The **Function Arguments** window will appear (*see image below*).

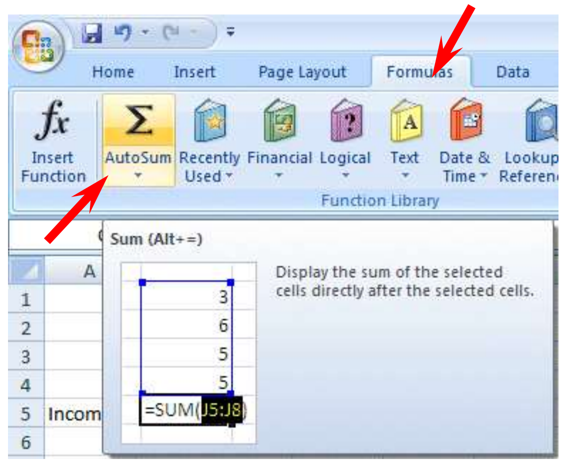


Highlight cells **C6 to C8** in the spreadsheet (*click-on C6, hold down the left mouse button, and drag until the three cells are highlighted*). A marquee will flash around the cells, indicating they are highlighted (*left arrow above*). The **Function Arguments** area will appear as above. Now

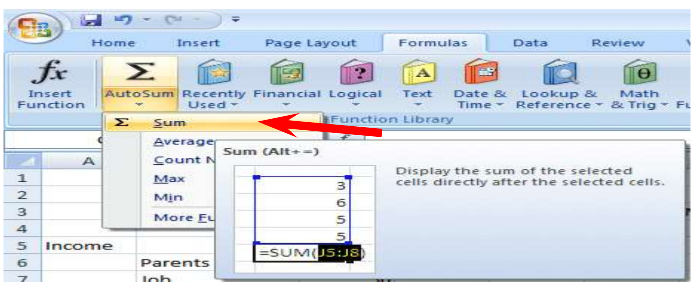
click the small button on the right of the cell (*see right arrow above*). The numbers will show in the area to the right of Number 1. Click **OK** at the bottom of the **Function Arguments** menu screen. You'll see that the **SUM** formula **[=SUM(C6:C8)]** shows in the formula area at the top of the screen. This is recommended method to highlight a group of numbers you want to add.

2.4.1 AutoSum METHOD - ∑

Since we add numbers more than any other operation in spreadsheets, Excel spreadsheet has an additional feature **Auto Sum**. Move to cell **C9** again and tap the **Delete** key to erase your last formula.



You should still be on the **Formulas Tab/Ribbon.** ***Notice (∑) Auto Sum** button.* Click the **AutoSum** button.

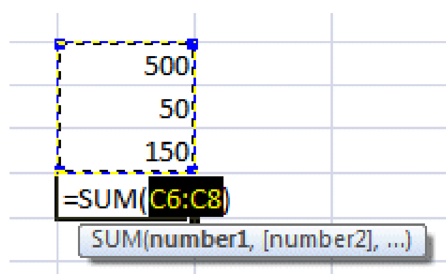


An image similar to the one on the left will appear.

Click ∑ **Sum.**

WOW!!! Automatic addition!!

Notice that the cells, you'd logically desire to add, have a marquee around them and that the **SUM** function is displayed in cell **C9**. You'll need to confirm that this is the **correct** formula. So tap the **Enter** key**,** and the SUM function will now be set in cell **C9**. Anytime you want to add using this method just click on the cell where you desire the total to be and click ∑ **Sum.**



This would be an appropriate time to save your work.

PERIODICALLY SAVE AND REPLACE YOUR WORK IN CASE YOU LOOSE POWER TO YOUR COMPUTER

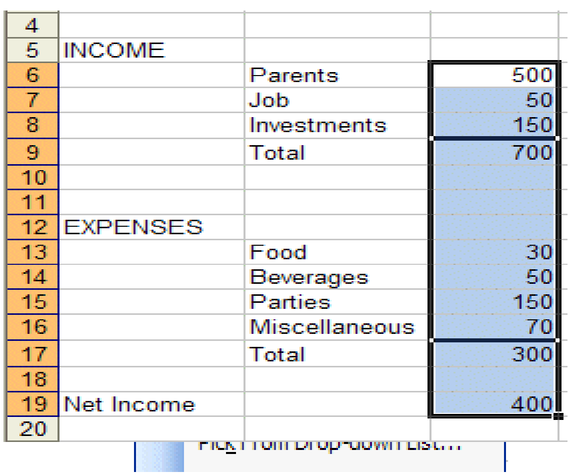
Now move to cell **C17** and add the total Expenses in cells **C13** to **C16** using each of the four methods. While you are in cell **C17**, go ahead and place a line at the top of cell **C17** using the format cells border **method** that you learned on **Page 5.**

2.4.2 Subtraction

In cell **A19** type-in **Net Income**. Next, adjust the width of column A (*as you did earlier at the introductory notes of this section*). Click on cell **C19**.
In cell **C19** we want to **subtract** (**-**) the amount in for **Expenses** in cell **C17** from the amount for **Income** in cell **C9**. This can be accomplished by using either the **Type-In** Method or **Point Method**. Go ahead and do this. Don't forget to tap the **Enter** key to confirm your formula.
The formula should look like **=C9-C17**

2.4.3 More Cell Formatting

We want our numbers to look better. To do this we'll include **dollar** signs and **decimal points** in our numbers. This is done by using the mouse. Point to cell **C6**, hold down the left mouse button and drag (move) down slowly to highlight cells **C6** through **C19**. Your screen should look like the image below.

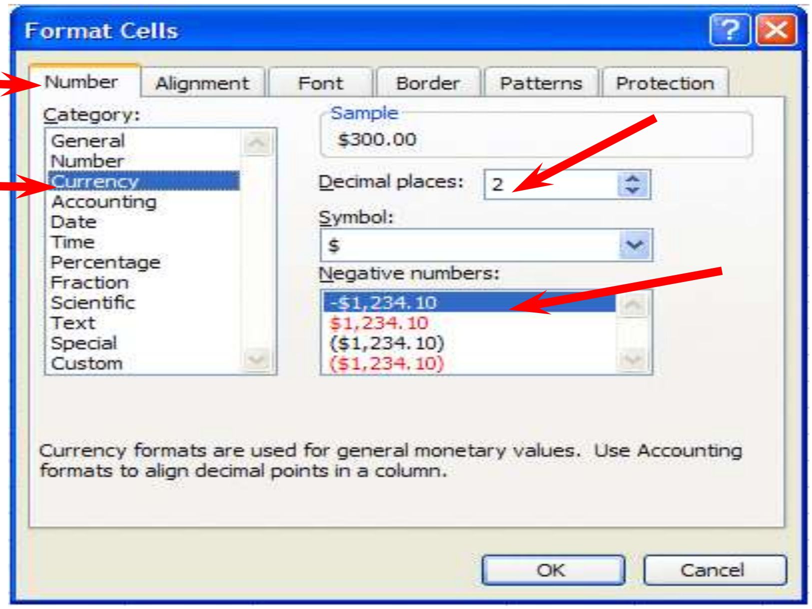


4			
5	INCOME		
6		Parents	500
7		Job	50
8		Investments	150
9		Total	700
10			
11			
12	EXPENSES		
13		Food	30
14		Beverages	50
15		Parties	150
16		Miscellaneous	70
17		Total	300
18			
19	Net Income		400
20			

Now point anywhere in the highlighted **area** and click the right mouse button. A pop-up menu will appear. Click on **Format Cells** (*like you have done before*).

Your Format Cells menu screen will appear and it will be similar to the image beside.

Click on the **Number Tab** at the top of the **Format Cells** menu screen. Point to **Currency** and click on **Currency.**



Notice several things:

- The right side shows the number of **decimal places.** The **2** is the default for cents. So we'll use **2**.

- The **Decimal Places** section, is a sample of what our number will look like.

- At the lower right it shows how **negative numbers** can appear**,** depending on your choice. When a negative number is calculated, it will appear in your desired format.

Now click on **OK**. All the numbers now have **$**. If you have large numbers that are too wide for the current column width, you will see some ######## in the cells where these numbers are to be located. If this occurs in your spreadsheet, go ahead and widen the columns as you did previously.

4			
5	INCOME		
6		Parents	$500.00
7		Job	$50.00
8		Investments	$150.00
9		Total	$700.00
10			
11			
12	EXPENSES		
13		Food	$30.00
14		Beverages	$50.00
15		Parties	$150.00
16		Miscellaneous	$70.00
17		Total	$300.00
18			
19	Net Income		$400.00
20			

Your spreadsheet numbers should now look like the one on the left.

`

2.4.4 Division

Now move to cell **A21** and type in the word **Percent.** We're going to calculate an arbitrary percentage to show you how division works and give you some more practice with numbers.

Now move to cell **C21.** Using either the **Type-In Method** or the **Point Method,** divide (/) the amount for **Income** in cell **C9** by the amount for **Expenses** in cell **C17**.

The **formula** should look like **=C9/C17**

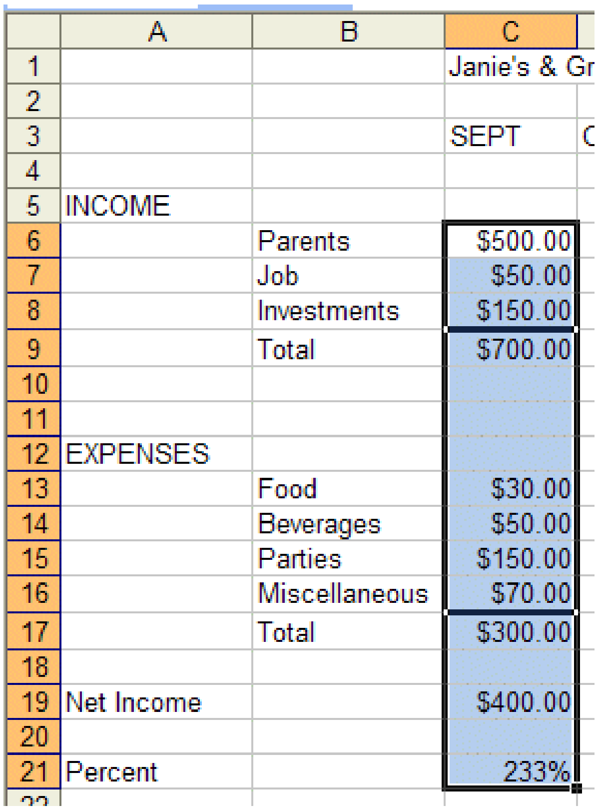
This will give you a horrid number, so why not attach a **percent symbol** to it. Now we'll repeat what we did above to format our **Currency ($$$$).**

2.4.5 Percentages

Point to cell **C21** and click the right mouse button. Point to **Format Cells,** then click the **Number tab,** then click on **Percentage.** Select **zero (O) Decimal Places**. Click **OK**.
You got it!!! **A %**.

Your spreadsheet should be similar to the image below.

2.4.6 Copying

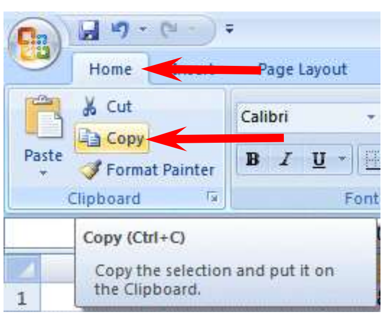


We could repeat what we did to this point and fill in the Income and Expenses for each of the remaining columns (months). There is a simpler way to do this.
Assuming our income and expense **amounts** are almost the same**,** throughout the months, we want to copy the amounts in **Column C** to **Columns D, E and F**. This will require **two steps**.

Firstly, Move your cursor to cell **C6**. We'll highlight what we want to copy.

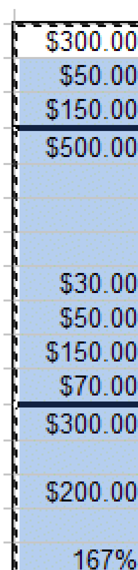
Secondly, we'll tell the spreadsheet where we want to place what we've copied.

So point to **C6,** hold down the left mouse button and drag (move) down the column until cells **C6** through **C21** are highlighted. Your highlighted area should look like the one on the left.

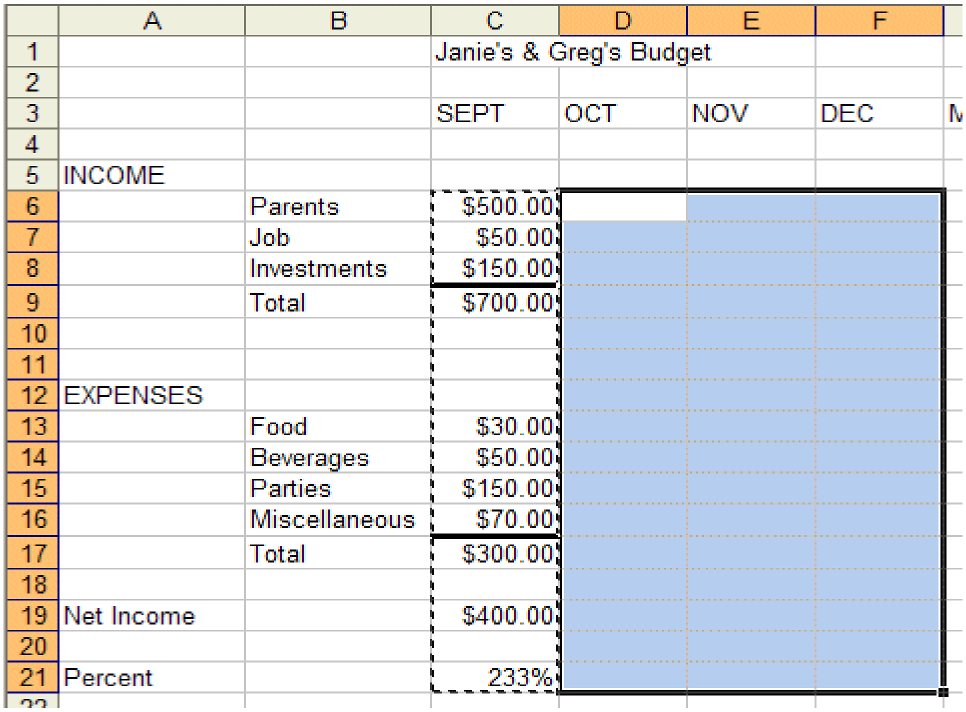


Click the **Home Tab** then **click** the **Copy** button.

You will notice that once again, when you highlight an area, a marquee of running lights moves around the **copy area**. So, you'll know you highlighted the correct area (*image on right*).

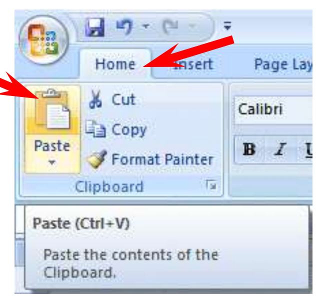


Now we'll tell Excel where to copy the data. Point to cell **D6,** click and hold down the left mouse button and drag down and to the right to cell **F21** (*This will highlight three columns: **OCT, NOV, DEC** to copy to.*). When you have finished highlighting, your screen should look like the image below.



Make sure you are still on the **Home Tab** and click the **Paste button**.

Great! All those numbers, dollar signs and formulas was copied in a flash!!



Your spreadsheet should not be different from the image on the right.

Click on a cell away from the area where the numbers are located. This will turn-off the highlight.
Or simply tap the **Esc key** and the marquee will disappear**.**

INCOME					
	Parents	$500.00	$500.00	$500.00	$500.00
	Job	$50.00	$50.00	$50.00	$50.00
	Investments	$150.00	$150.00	$150.00	$150.00
	Total	$700.00	$700.00	$700.00	$700.00
EXPENSES					
	Food	$30.00	$30.00	$30.00	$30.00
	Beverages	$50.00	$50.00	$50.00	$50.00
	Parties	$150.00	$150.00	$150.00	$150.00
	Miscellaneous	$70.00	$70.00	$70.00	$70.00
	Total	$300.00	$300.00	$300.00	$300.00
Net Income		$400.00	$400.00	$400.00	$400.00
Percent		233%	233%	233%	233%

Change a few numbers in each of the **months** specifically under the **income** and **expense** areas to see how the spreadsheet works.

You will notice how all of the formulas, totals, and percentages change automatically. This is the most exciting feature of an electronic spreadsheet.

This will make the graphs we'll create be more realistic later in the tutorial.

Our spreadsheet now looks like the image on the right.

You may now save your work.

	A	B	C	D	E	F
1			Janie & Greg's Budget			
2						
3			SEPT	OCT	NOV	DEC
4						
5	Income					
6		Parents	$500.00	$500.00	$500.00	$1,000.00
7		Job	$50.00	$50.00	$50.00	$200.00
8		Investments	$150.00	$20.00	$150.00	$150.00
9		Total	$700.00	$570.00	$700.00	$1,350.00
10						
11						
12	Expenses					
13		Food	$30.00	$100.00	$30.00	$200.00
14		Beverages	$50.00	$100.00	$50.00	$200.00
15		Parties	$150.00	$150.00	$150.00	$500.00
16		Miscellaneous	$70.00	$70.00	$70.00	$70.00
17		Total	$300.00	$420.00	$300.00	$970.00
18						
19	Net Income		$400.00	$150.00	$400.00	$380.00
20						
21	Percent		233%	136%	233%	139%

Entering formulas in the Monthly Totals Column

Click cell **G6** (*under the title **Monthly Totals***). Choose one of the formulas you learned earlier to add the four monthly amounts in the **Parents** row (*Use any of the four methods you desire)*. Your spreadsheet should be similar to the image below:

	A	B	C	D	E	F	G	H
3			SEPT	OCT	NOV	DEC	MONTHLY TOTALS	
4								
5	Income							
6		Parents	$500.00	$500.00	$500.00	$1,000.00	=SUM(C6:F6)	
7		Job	$50.00	$50.00	$50.00	$200.00	SUM(number1, [number2], ...)	

After you have added the four **columns** in **cell G6**, you'll **copy** the formula in cell **G6** to cells **G7** through **G19**. Click on cell **G6** and follow the **Copy** process you did some few topics before.

Next, click on **cell G7**, hold down the left mouse button, and drag down through cell **G19**. Your spreadsheet should look like the one to the right.

F	G
DEC	MONTHLY TOTALS
$300.00	$1,200.00
$50.00	
$150.00	
$500.00	
$30.00	
$50.00	
$150.00	
$70.00	
$300.00	
$200.00	

Now follow the **Paste process** you used (on page 80) to paste the formula from cell G6 to cells G7 through G19.

MONTHLY TOTALS
$1,200.00
$200.00
$600.00
$2,000.00
$0.00
$0.00
$0.00
$120.00
$200.00
$600.00
$280.00
$1,200.00
$0.00
$800.00

After you’ve pasted your formula, you will see some zeroes in cells **G10, 11, 12, and 18**. This is because there was nothing to add in that area. So, go-in and clean-up these cells by deleting the zeros in these cells.

Copying the Percentage Formula

Notice that we didn’t copy the percentage formula when we did the last copying process. If we had copied a SUM formula, it would have added the four percentages. We don’t want the sum of the percentages. Rather we want a percentage fomula which only applies to the overall **Monthly Totals**. So, we need to copy the percentage formula separately. Click on cell **F21**, copy the percentage formula in cell **F21** to cell **G21**. This is the average percentage that **Income** is greater than **Expenses**.

Now put a **$** in cells **G6 through G19** (*like you did earlier*), and a **%** in **G21**. Your spreadsheet column **G** should be similar to the image on the right

*You can now **save** you work before proceeding to the next stage.*

MONTHLY TOTALS
$1,500.00
$200.00
$750.00
$2,450.00
$380.00
$200.00
$655.00
$310.00
$1,545.00
$905.00
159%

2.4.7 Absoluting (and multiplication)

There are times when we are working with a spreadsheet and we do not want a cell to *roll* to the next column when we use the copy feature of the spreadsheet, like it did in our last copying exercise. To stop the cells from *rolling* we utilize something called ***absoluting***. The following is an illustration of absoluting.

Go to cell **A23** and type-in **Number**. Go to cell **A25** and type-in **Result**.

Then go to cell **C23** and type-in the number **2**, tap the **Enter** key.

We'll now create a formula to multiply our **number** by **Net Income**. You may use either the **Type-in** or **Point method**. Go to cell **C25,** and type-in a formula to multiply cell **C23** by cell **C19**.

The formula should look like **=C23*C19**

The result in **C25** should be two times the **net income** in cell **C19**.

Now **copy** the formula in cell **C25** to cells **D25, E25, F25 and G25**. Your row 25 should look similar to the one below.

23	Number		2				
24							
25	Result		$800.00	$0.00	$0.00	$0.00	$0.00

Uh oh! Where did all of those "0's" come from?

Point to each of the cells **D25, E25, F25 and G25**. Notice, as you click on each cell, and look at the screen, how **C23** (*the cell with the 2*) **rolled and became D23, E23, F23 and G23** (*which are blank as a result of the **0s***). A blank cell multiplied by a number is a 0. We want the 2 to be in each formula and not to **roll.**

To do this we utilize something called ***Absoluting*** *or* ***Anchoring***.

Go back to cell **C25**. Now we'll enter the formula again, but a little differently (*to anchor the 2*).

Type-in **=C23** (*or you could type = and point to C23*). Now, tap the F4 function key. **Notice**, in cell **C25** and the **Edit** bar at the **top of the screen** that the **=C23 changes to: C23**. (*This tells you that cell **C23** is **absoluted** or **anchored**. The **$s** indicate the **absoluting**.*) Now finish the formula by typing in or pointing ***C17** as before. Tap **Enter**.

The formula in cell C25 should look like: =**C23*C19**

Now copy the formula in cell **C25** to cells **D25, E25, F25 and G25** again. Your row 25 should be similar to the image below.

23	Number		2				
24							
25	Result		$800.00	$0.00	$0.00	$0.00	$0.00

The numbers should now be correct. Point to cells **D25, E25, F25 and G25** (*like you did before*). You will notice the **$s** have copied the **=$C$23** to each cell **(***absoluting***)** and the **Net Income** figures (***Cells D19, E19, F19 and G19** have **rolled** as they should. Absoluting is something you should know and understand.)*

> ***Pause and reflect:** Look at all you have accomplished. If you want go-in and change some more numbers or change the **income** and **expense** titles to something you feel is more fun or appropriate, you are free.*

You can now save your work and move to the next section.

*The next important lesson is to learn how to **print.** This done with a few and easy steps.*

2.5 Printing

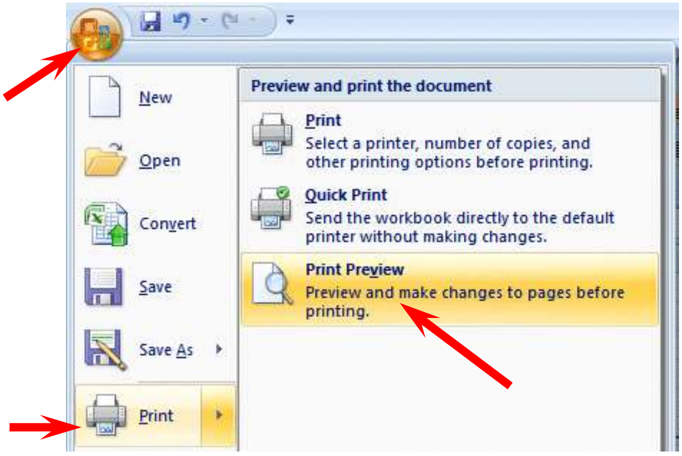
First, click cell **A1**.

All of the Windows in the spreadsheet try to figure out what you want to print. They can be right and the otherhand wrong. So the most important thing to note about printing, is to tell the printer what to print, rather than it suggesting it for you.

Unlike a word processor, you may need to highlight what you want to print. For the moment, we'll assume that Excel 2007 will guess correctly, and that you have not clicked somewhere that will cause a problem. If you do have problems, which we'll know in a moment not long from now. We'll equally show you how to take care of the problem.

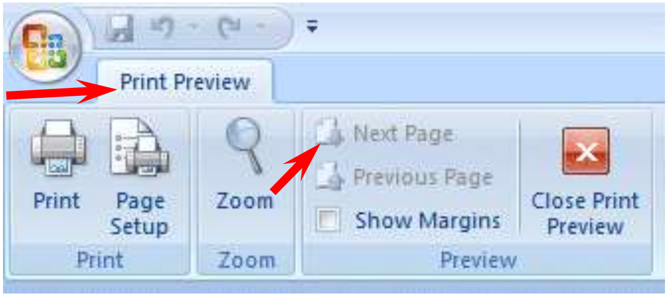
It's usually a good idea to see what our printout will look like before you print it. First, we'll use a **Print Preview** to **see** what our spreadsheet looks like.

Click the **Microsoft Office Button**.



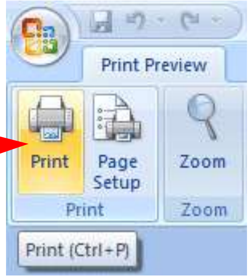
When the menu screen appears, move your cursor over **Print** and then click the **Print Preview** choice.

At the top of the **Print Preview** screen you will see the **Print Preview Tab**.



We'll use these buttons to assist us with our printing.

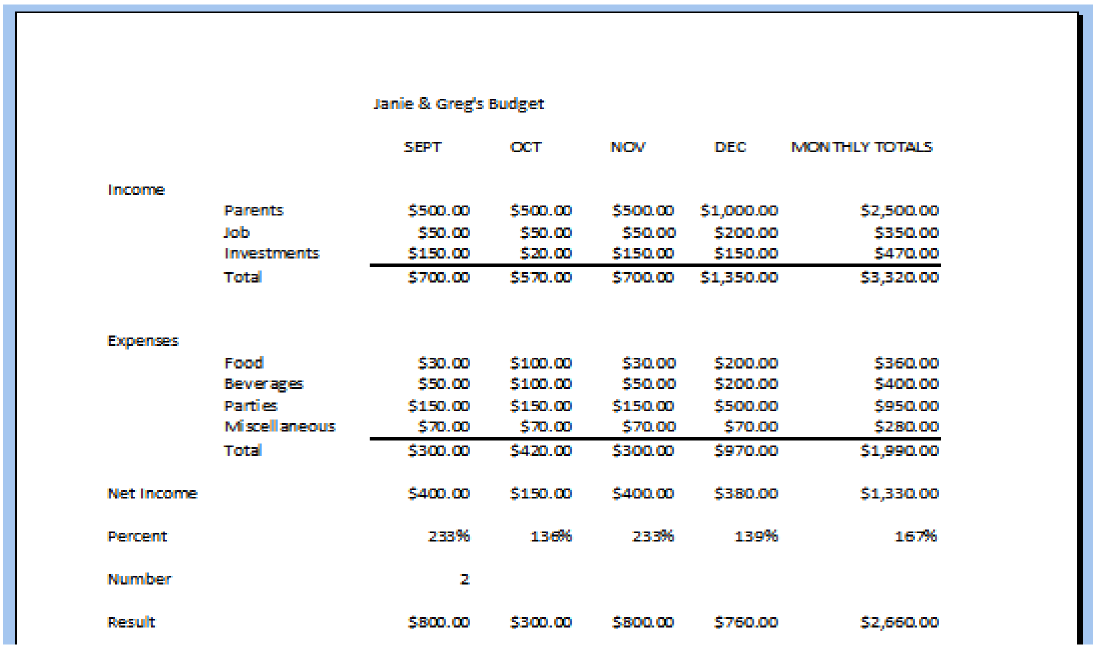
Notice: the **Next** and **Previous** buttons are not highlighted - they are just gray. This means that the buttons are *not active* and indicates that we are OK with our spreadsheet – it is all on one page. If the Next button was active, this could imply that there are other pages to our spreadsheet. Look at the lower left corner of the **Print Preview screen** you'll see**, Preview: Page 1 of 1.** This confirms that our spreadsheet is on one page. If you do not see this combination, we'll show you how to take care of it later.



If you do see this combination, click the Print button. Then, click on **OK** in the **Print menu** screen that appears.

Label (*write on*) this **printout**: **Default Spreadsheet Printout**.

An image of the printout will appear and should be similar to the image below.

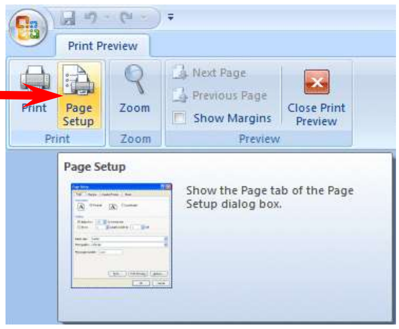


Janie & Greg's Budget

		SEPT	OCT	NOV	DEC	MONTHLY TOTALS
Income						
	Parents	$500.00	$500.00	$500.00	$1,000.00	$2,500.00
	Job	$50.00	$50.00	$50.00	$200.00	$350.00
	Investments	$150.00	$20.00	$150.00	$150.00	$470.00
	Total	$700.00	$570.00	$700.00	$1,350.00	$3,320.00
Expenses						
	Food	$30.00	$100.00	$30.00	$200.00	$360.00
	Beverages	$50.00	$100.00	$50.00	$200.00	$400.00
	Parties	$150.00	$150.00	$150.00	$500.00	$950.00
	Miscellaneous	$70.00	$70.00	$70.00	$70.00	$280.00
	Total	$300.00	$420.00	$300.00	$970.00	$1,990.00
Net Income		$400.00	$150.00	$400.00	$380.00	$1,330.00
Percent		233%	136%	233%	139%	167%
Number		2				
Result		$800.00	$300.00	$800.00	$760.00	$2,660.00

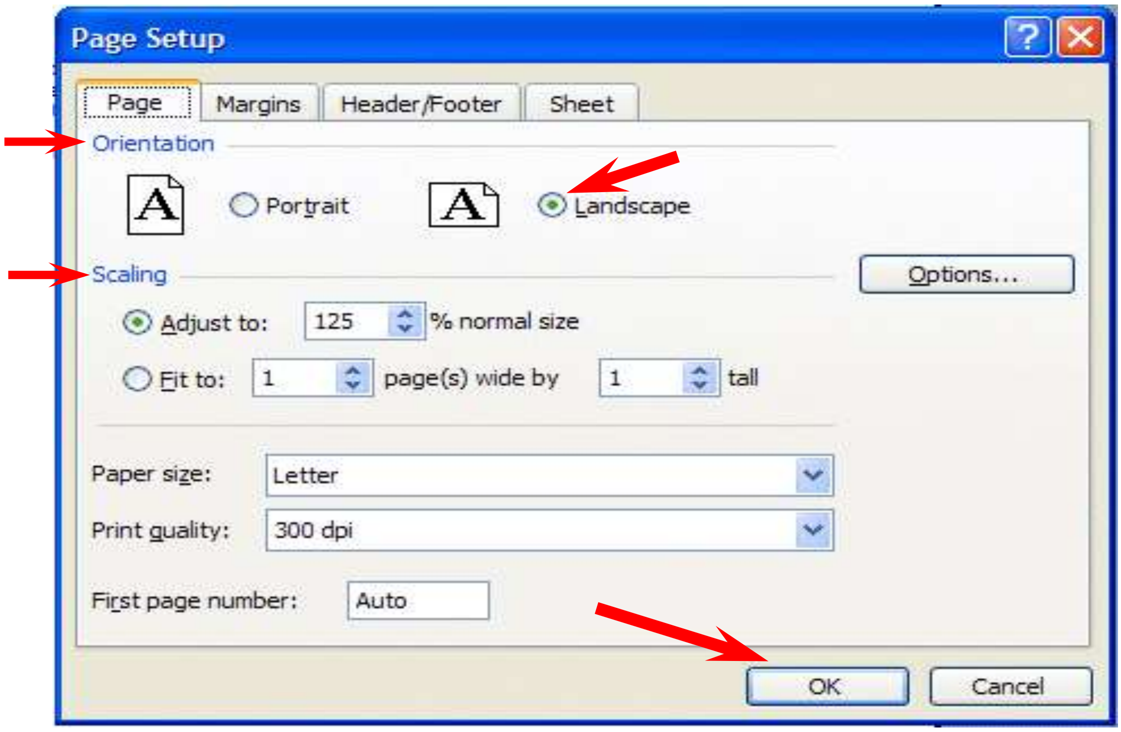
The image above should also be identical to your spreadsheet.

If you move your cursor over the spreadsheet, you'll notice that the cursor changes from an **arrow** to a **tiny magnifying glass**. If you click the left mouse button, your **magnifying glass will** zoom-in on the spot where the magnifying glass is located. If you click again, it will zoom-out. Try this a couple of times. It is a handy feature.



Now **click** the **PageSetup button** on the **Print Preview Tab**.

The **Page Setup** menu screen at the top of the next page will appear.



Notice that the **Page Setup** menu screen indicates that you are in **Portrait** view. Now we'll enhance the spreadsheet to make it a bit more presentable. In the **Orientation** area click in the small circle to the left of **Landscape (***see arrow above***)**. The spreadsheet will now print on the page as indicated. Next, in the **Scaling** area, click in the box to the left of **% normal size**. Using either the **up/down arrows** or **typing**-in the information, change the size to **125**. Then click **OK**.

Your spreadsheet will now be larger and fill the paper more appropriately. Click on **Print** and when this spreadsheet comes out of the printer label it: **Landscape – enlarged to 125 %**.

Go ahead and adjust the size of your spreadsheet so that it becomes too large to fit on a single page. Set the **Scaling** to **200** and click **OK**. When you return to the **Preview** screen, the **Next** and **Previous** buttons at the top left will now be **active**, and you'll see **1 of 3 or 4 pages** in the **lower left corner** of the screen. Go ahead and click the **Next** and **Previous buttons** to get a "feel" for the "size" of your spreadsheet. If you click on Print (*please don't do it*), you'll get these 3 or 4 pages. If you made a mistake when you created the spreadsheet, you might see that you have 58 *(some larger number of*) pages in your spreadsheet!

Now, click in the small circle to the left of **Fit to 1 page(s) wide by 1 tall** in the **Scaling** area and make sure that **1 page** is set. Excel 2007 will now return your spreadsheet to **one** page. Try other things here. Work with the **Margins, Header/ Footer, and Sheet** tabs at the top of the Page Setup menu screen. Any time you desire to print, go ahead and do so. This will give you an insight of how the spreadsheets will print. When you are finished, simply click **OK** or **Cancel** and you will return to your spreadsheet.

Many people ask:
- How to center a spreadsheet on the page? This feature is located on the **Margins** tab at the bottom left of the **Margins** screen.

- How to place **gridlines** and show the row and column headings (A, B, C and 1, 2, 3) in their spreadsheet printouts? This feature is located on the **Sheet tab** in the Page Setup menuscreen.

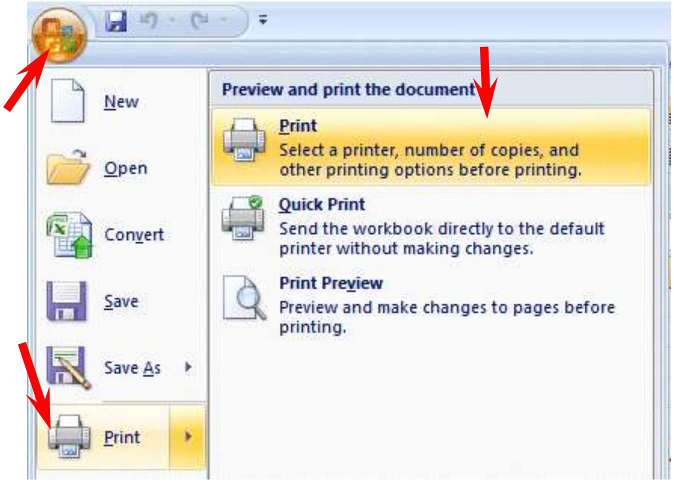
When you are finished working with Print Preview, click the **Close Print Preview** button. When you return to your spreadsheet you will see dashed lines around your data. **Print Preview** added these to assist you in knowing where the edges of your printed data will be on paper. We'll show you another nice feature that might assist even more shortly.

if you have many spreadsheet pages.

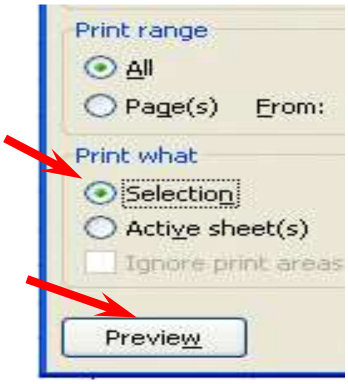
Click cell **A1** and **highlight** your spreadsheet down through cell **G25**.

After you've highlighted **A1** through **G25,** click the **Microsoft Office** Button and then click **Print**. In the **Preview and print the document** area (*on the right*), click **Print**

A **Print menu** screen will appear.



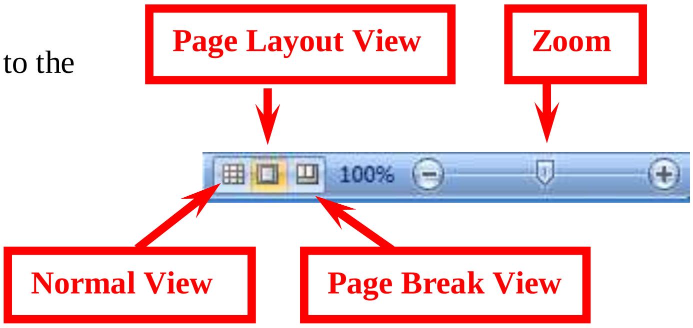
In the lower left corner of the **Print menu** screen you will see an area that looks like the image on the right. Click in the small circle to the left of **Selection**. This indicates to Excel that you only want to print the area you've highlighted. Click **OK**. Only the section that you've highlighted will print**.** You can still modify your spreadsheet if you desire. Once you've clicked **Selection**, you may click on the **Preview** button to see a preview of your highlighted area. Follow the instructions above to modify as you desire.



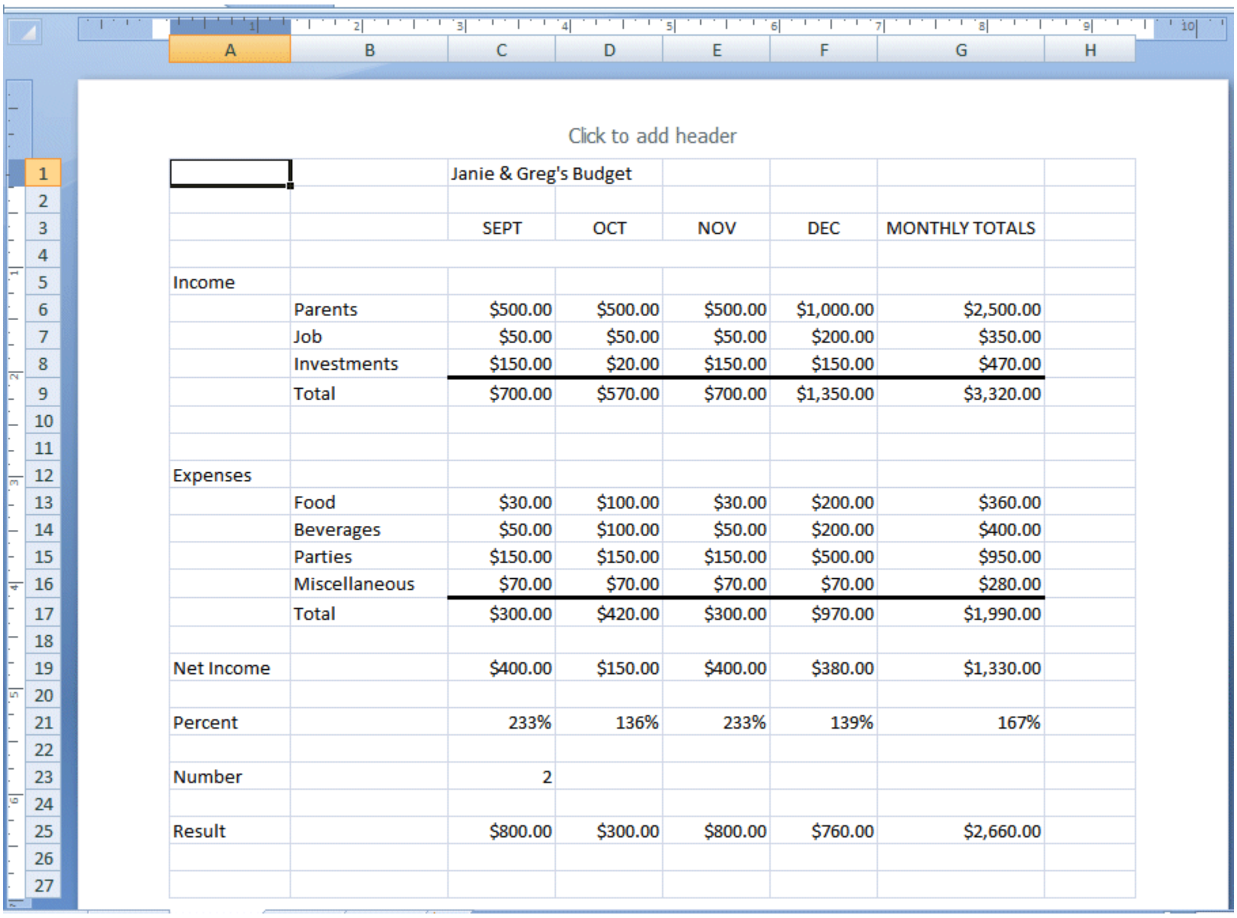
2.5.1 Page Layout View

Now that you have an insight for printing your spreadsheets, we'll look at another feature in Excel 2007 – **Page Layout View**. In the lower right corner of your **Excel screen** you'll see the Excel 2007 Toolbar.

The **Excel View Toolbar** looks similar to the image on the right.



You are currently in **Normal View**. Look at your screen and then click the **Page Layout button** (*as shown above*) on the **Excel View Toolbar**. Your Excel screen should look similar to the image below.



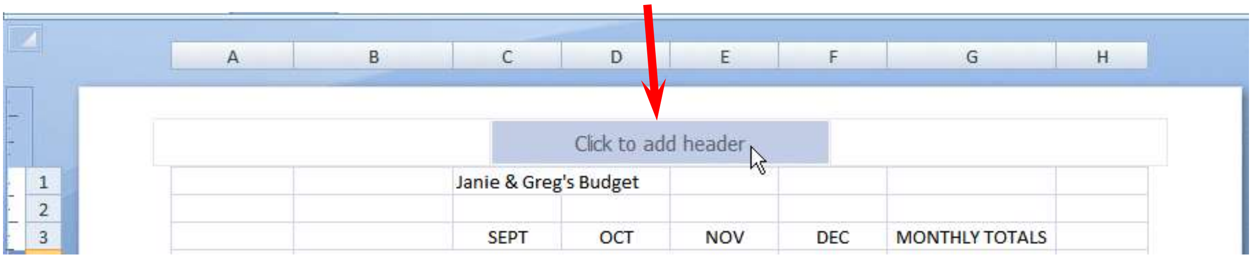
		Janie & Greg's Budget				
		SEPT	OCT	NOV	DEC	MONTHLY TOTALS
Income						
	Parents	$500.00	$500.00	$500.00	$1,000.00	$2,500.00
	Job	$50.00	$50.00	$50.00	$200.00	$350.00
	Investments	$150.00	$20.00	$150.00	$150.00	$470.00
	Total	$700.00	$570.00	$700.00	$1,350.00	$3,320.00
Expenses						
	Food	$30.00	$100.00	$30.00	$200.00	$360.00
	Beverages	$50.00	$100.00	$50.00	$200.00	$400.00
	Parties	$150.00	$150.00	$150.00	$500.00	$950.00
	Miscellaneous	$70.00	$70.00	$70.00	$70.00	$280.00
	Total	$300.00	$420.00	$300.00	$970.00	$1,990.00
Net Income		$400.00	$150.00	$400.00	$380.00	$1,330.00
Percent		233%	136%	233%	139%	167%
Number		2				
Result		$800.00	$300.00	$800.00	$760.00	$2,660.00

*Great! Your screen now has all the settings you entered in **Print Preview**. Notice the Rulers at the top and on the left just like Microsoft Word. So you now have an accurate Print Preview.*

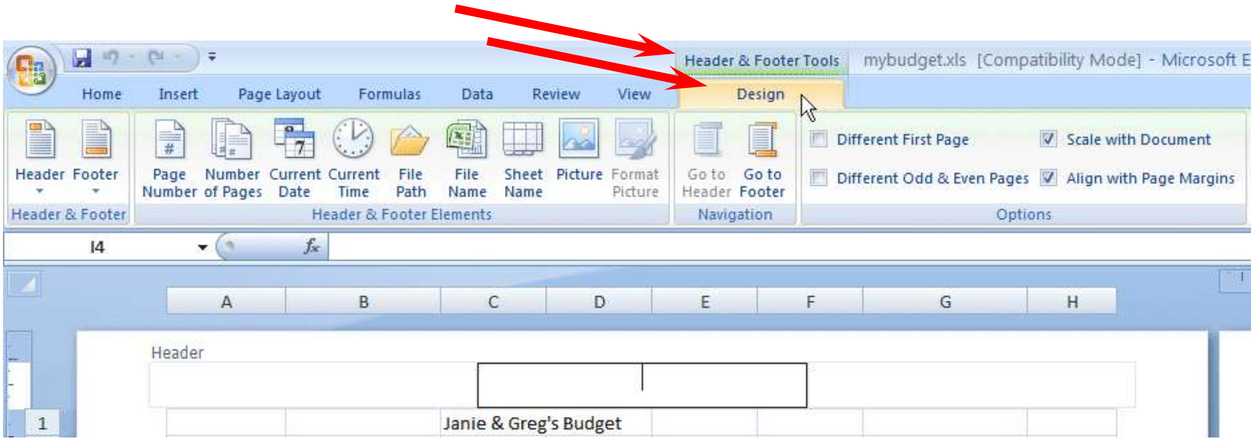
Spend a few minutes using the **Elevator Bars** on the Right side and Bottom of your Excel spreadsheet to move up and down and left to right. As you move, you'll see the Page Breaks and Pages as they'll be printed!

2.5.2 Headers and Footers

Look at the top of your spreadsheet and you'll also see an area which indicates: **Click to add header**.



If you desire a **Header** (or **Footer**) on each spreadsheet page, you can now create them in this view. **Notice** (*above*) we've moved our cursor over the **center Header area** (*the Footer area is at the bottom of the page*). When we did it **turned light blue**. If you move your cursor over the left and right Header areas, you will see that they'll turn blue as well. If you then **click** on one of these areas you'll **see** a **Header & Footer Tools Tab** – with a **Design Tab below**. In the **Design Ribbon** you'll see that this Tab/Ribbon is "tailored" to work with creating your Headers and Footers. This is one of the great new features in 2007 Office and Excel.



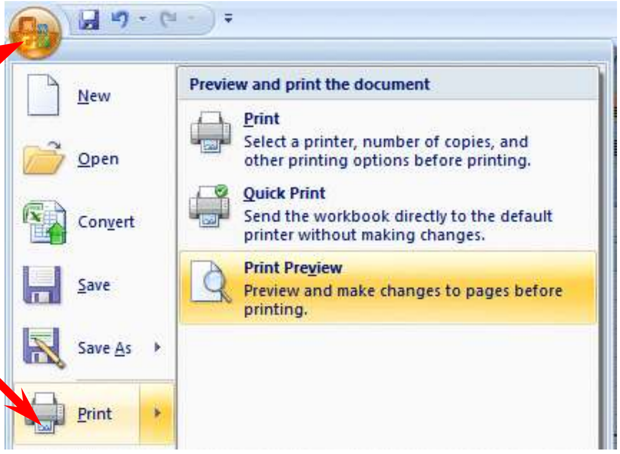
Go ahead and experiment as you desire. This new feature really makes working with Headers and Footers really easy.

Print Preview Button in Quick Access Toolbar

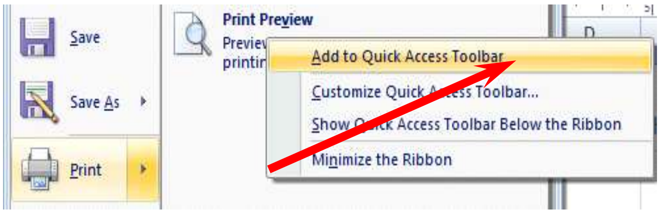
Since you'll be using the **Print Preview** feature frequently, it would be nice to have a button in the **Excel Quick Access Toolbar**, so you won't have to go through all the clicking.

To add a **Print Preview button,** click the **Microsoft Office** Button as we did earlier to open our Print Preview.

When the menu screen appears, move your cursor over **Print** and then move the cursor over the **Print Preview** choice.

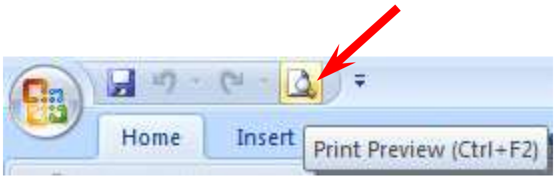


When the **Print Preview** selection turns orange, click the right mouse button and a pop-up menu will appear.



Move your cursor over the **Add to Quick Access Toolbar** selection and click the left mouse button.

As soon as you click, you'll see your Print Preview button added to the Excel Quick Access Toolbar. Now, anytime you desire to Preview the output of your spreadsheet, all you'll have to do is, click the Quick Access Print Preview button.



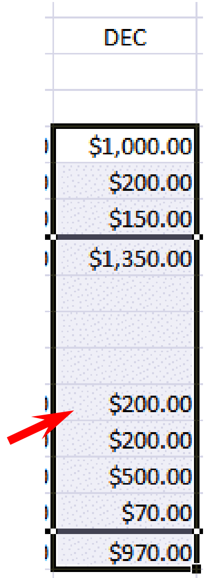
As you see other buttons you would like to add to your Excel Quick Access Toolbar, simply follow the instructions above.

2.6 Graphics

This is an exciting section. We are now going to turn your spreadsheet numbers into graphics - *bar charts*, *pie charts*, etc. This will greatly assist you when you display and explain your work. A new really awesome feature in Excel 2007 is Conditional Formatting. We'll begin with Conditional Formatting and them move to full page Charts.

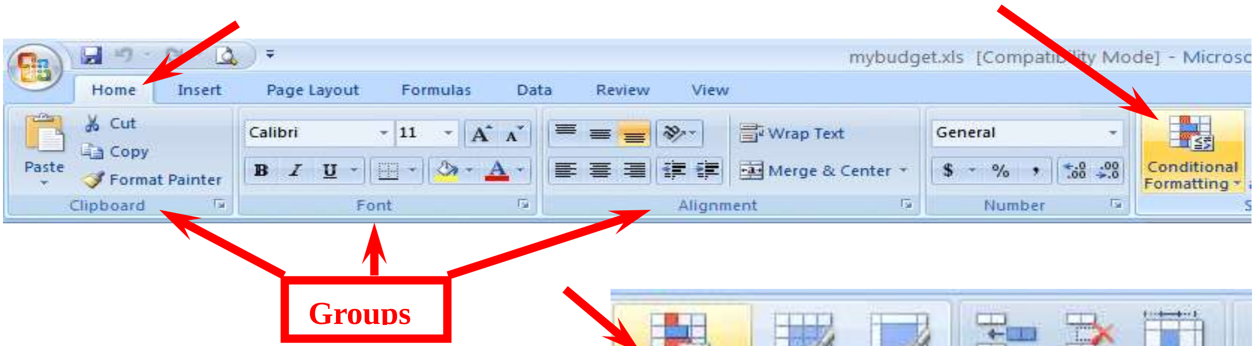
2.6.1 Conditional Formatting

Conditional Formatting will let you display graphics in your spreadsheets. The best way to describe this amazing feature is to show you how it's done.

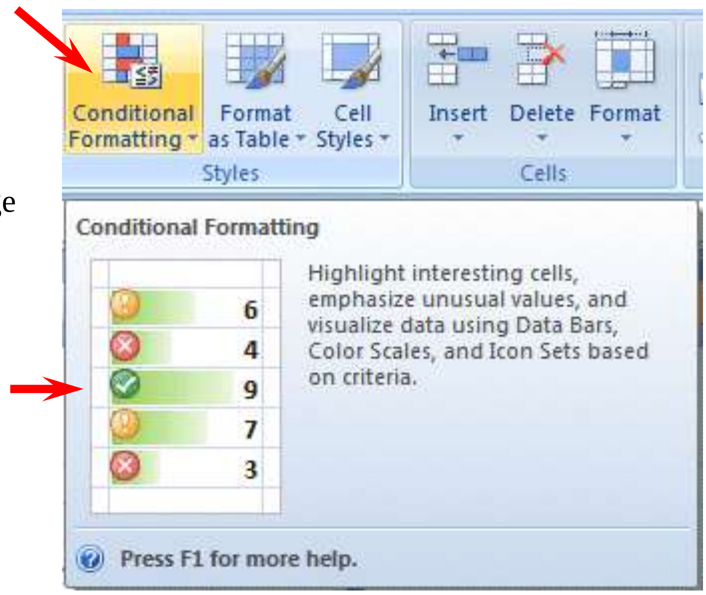


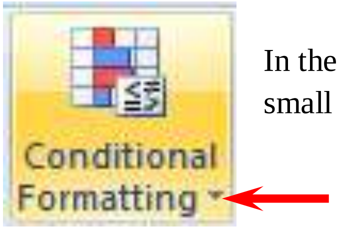
First, you'll need to highlight some of the data on your spreadsheet. We highlighted the **Income** and **Expense** numbers for the month of **December** for our image. Our Conditional Formatting graphics will appear in this column.

Now, look at the tabs at the top of your Excel screen and make sure you are on the **Home Tab**. Under each Tab is a Ribbon made up of Groups (*Clipboard*, *Font*, *Alignment*, *etc.*). Over to the right is a **Conditional Formatting** selection in the **Styles Group**.

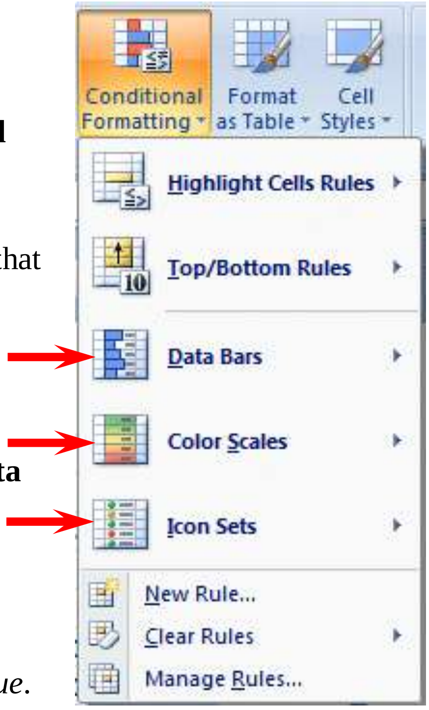


When you move your cursor over the Conditional Formatting button, an image similar to the one on the right will appear. We've enlarged the image so you can get an idea of how this feature will work.





In the lower right corner of the **Conditional Formatting button** is a small down arrow. Click on this **arrow**.

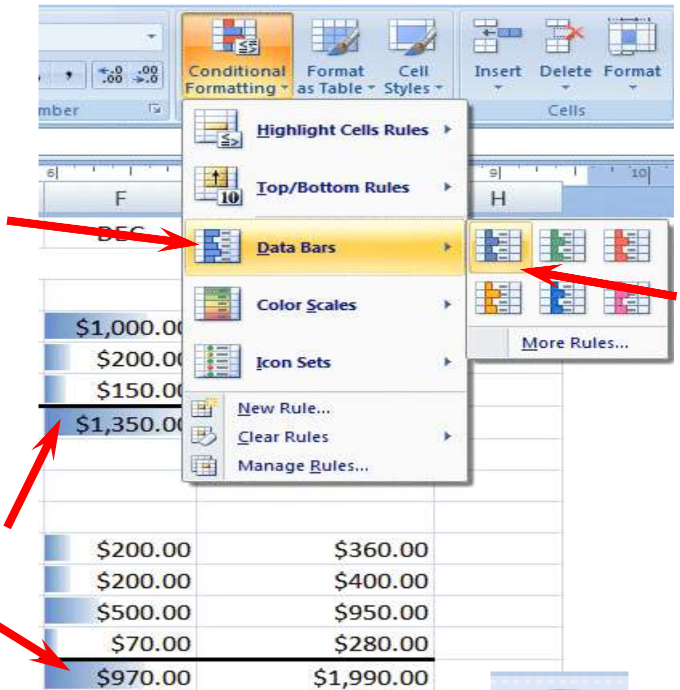


When you click the down arrow the image similar to the one beside will appear. **Notice** the **Data Bars, Color Scales and Icon Sets choices**.

As you move your cursor over these selections you will see that a number of choices available with each option.

To show you how this works, we moved our cursor over **Data Bars**.
Then, we moved our cursor over the
Blue selection on the right.

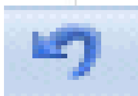
Notice, our December Column data is now highlighted in *blue*.



Also notice that the larger numbers have a longer blue bar highlight with them.

This is an interesting feature. Go ahead and look at **Color Scales** and **Icon Sets** as well. You can carryout several tests as you desire. If you click a selection it will display that graphic on your spreadsheet. If you go to **Print Preview**, you will see that effect. Now you can print spreadsheets that have not only data numbers, but a graphic to show their size!

If you don't like the Conditional Formatting effects, you can use the Undo Arrow (*in the Quick Access Toolbar*) to remove them.

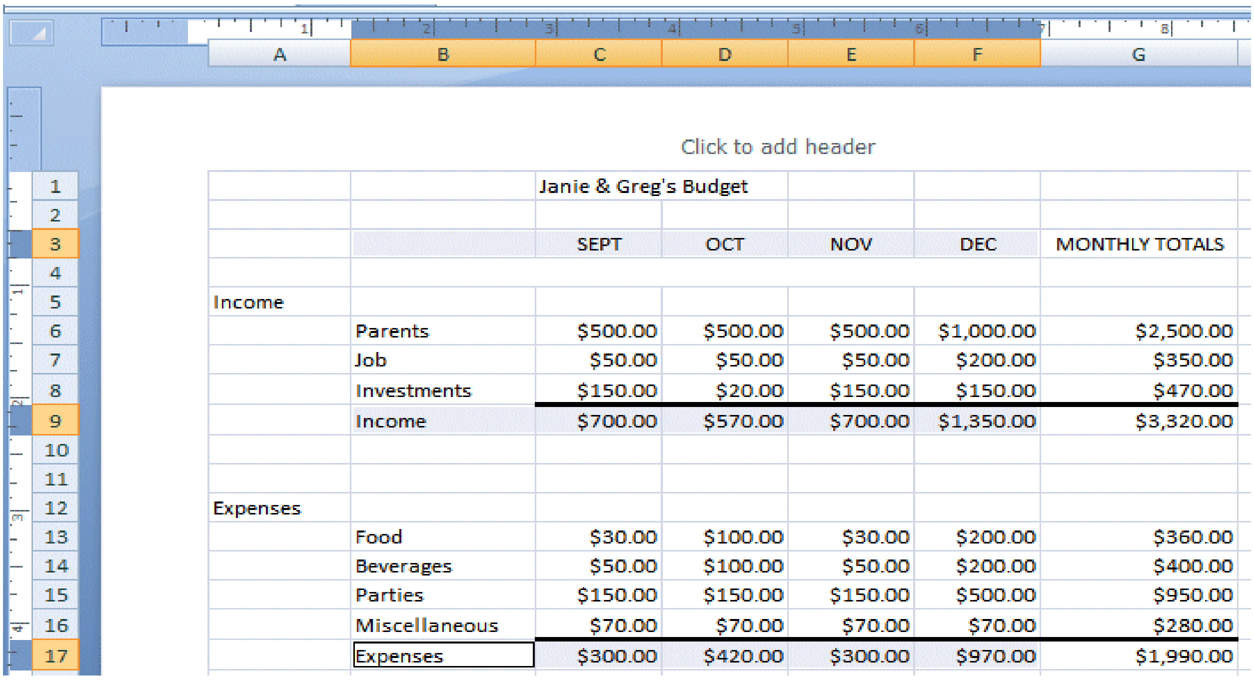


2.7 Charts

Before we work with Charts, there are a couple of essential steps we have to do. To begin, we have to tell Excel 2007 what we want to see in our chart and then where we want the chart to go.

First, in cell **B9** type-in **Income** (*to replace the word **Total***).
Next, type-in **Expenses** in cell **B17** for the same reason. You'll understand the changes we are doing when you see your chart.

Hold down the left mouse button and highlight cells **B3** to **F3**. Next, hold down a **Ctrl** key (*at the bottom of the keyboard)* and highlight cells **B9** through **F9**. You will now see two ranges highlighted. Hold down the Ctrl key again**,** and highlight cells **B17** through **F17**. These three ranges will make up your chart. The x-axis will be made up of cells **B3** through **F3**. And, the two sets of bars – Series - will show **Income** and **Expense**. When you complete the above instructions, your screen should look like the image below.



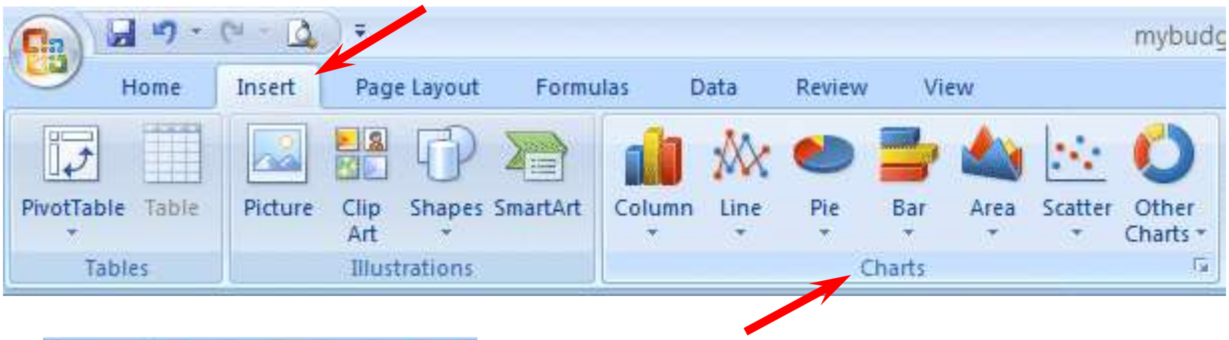
Click to add header

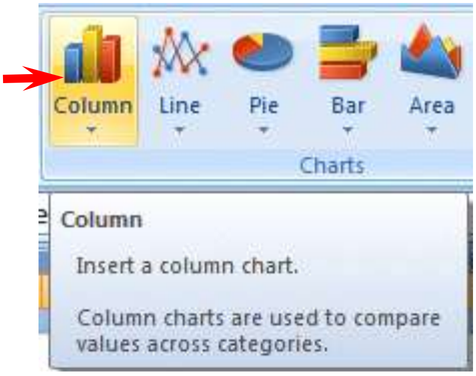
	A	B	C	D	E	F	G
1			Janie & Greg's Budget				
2							
3			SEPT	OCT	NOV	DEC	MONTHLY TOTALS
4							
5	Income						
6		Parents	$500.00	$500.00	$500.00	$1,000.00	$2,500.00
7		Job	$50.00	$50.00	$50.00	$200.00	$350.00
8		Investments	$150.00	$20.00	$150.00	$150.00	$470.00
9		Income	$700.00	$570.00	$700.00	$1,350.00	$3,320.00
10							
11							
12	Expenses						
13		Food	$30.00	$100.00	$30.00	$200.00	$360.00
14		Beverages	$50.00	$100.00	$50.00	$200.00	$400.00
15		Parties	$150.00	$150.00	$150.00	$500.00	$950.00
16		Miscellaneous	$70.00	$70.00	$70.00	$70.00	$280.00
17		Expenses	$300.00	$420.00	$300.00	$970.00	$1,990.00

If your spreadsheet does not look exactly like the one above, please try again. This is a bit tricky and it often takes (*even experienced spreadsheet users*) a couple of tries to get the highlighting just right.

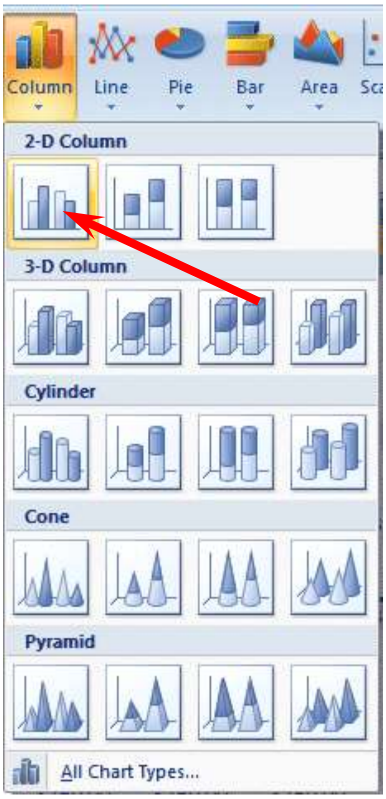
What we are about to do is new in Excel 2007. So, if you have used previous versions of Excel – hang on – this is awesome!

Click the **Insert Tab** at the top of the **Excel screen**. When you do you'll see that one of the Groups under the **Insert Tab/Ribbon** is **Charts**.





For our first chart, we'll use a **Column Chart**. Move your cursor over Column and an image like the one on the right will appear.

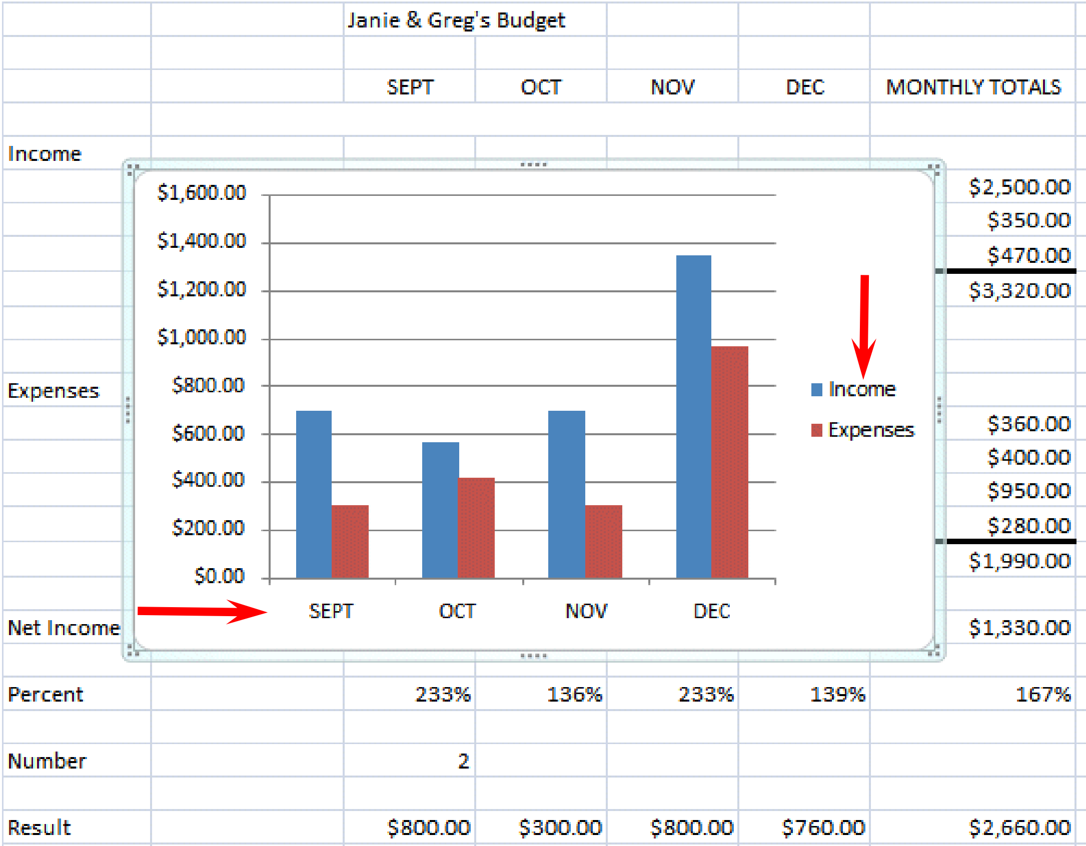


Click the **Column button** and you'll see an image like the one on the left. As you can see there all kinds of Column Charts. We'll begin with a simple two dimension (**2-D) Column** chart. You'll be able to change this later, (*if you desire*) to another Column Chart of one of the other selections.

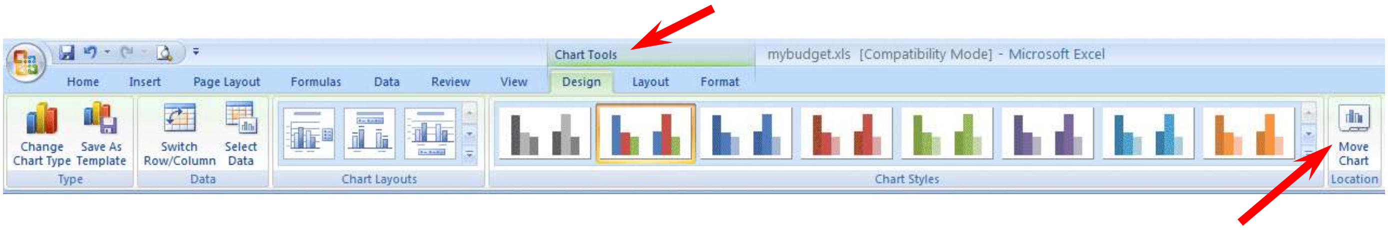
Click on the **2-D Column** chart indicated by the arrow on the right.

As soon as you click, because you highlighted your data, an image similar to the one below will appear.
Don't worry that the Chart is covering your data – we'll take care of that shortly.

Notice that when we highlighted cells **B3 to F3** this created the *X-Axis* labels (*SEPT, OCT, NOV etc.*). When we highlighted cells **B9 to F9** and **B17 to F17** this created the two **Income** and **Expense** bars for each month. And, when you typed **Income** into **B9** and **Expenses** into **F9** this created a **Legend** on the right side of your chart.

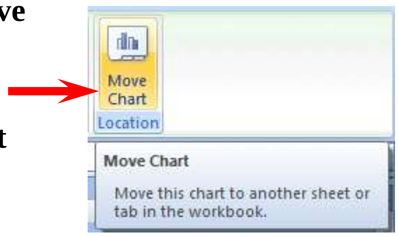


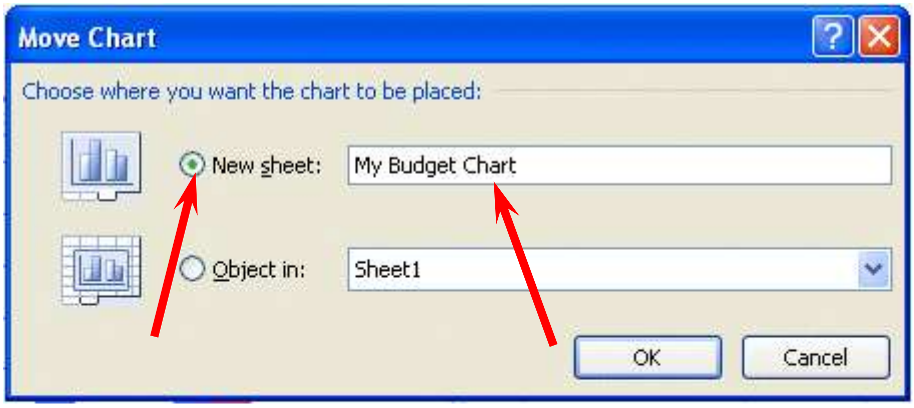
Now, let's move our chart to a page of its own – so it will be easier to work with and not cover our data. Click anywhere on your chart and you will notice that a new Tab appears at the top of your Excel screen – **Chart Tools**. Click the **Chart tools** Tab and the **Chart Tools Tab/Ribbon** will appear like the image below.



Notice, on the right end of the **Chart Tools Tab/Ribbon** is a **Move Chart Location** button. Click the **Move Chart Location** button.

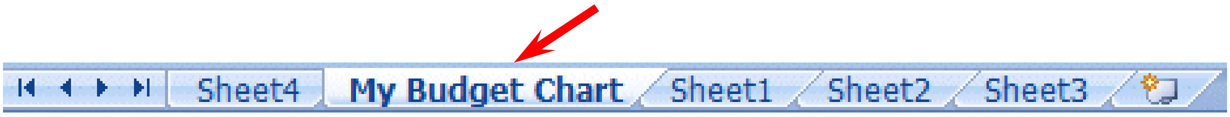
When you click the **Move Chart Location button** a **Move Chart menu screen** will appear (*like the image below*).



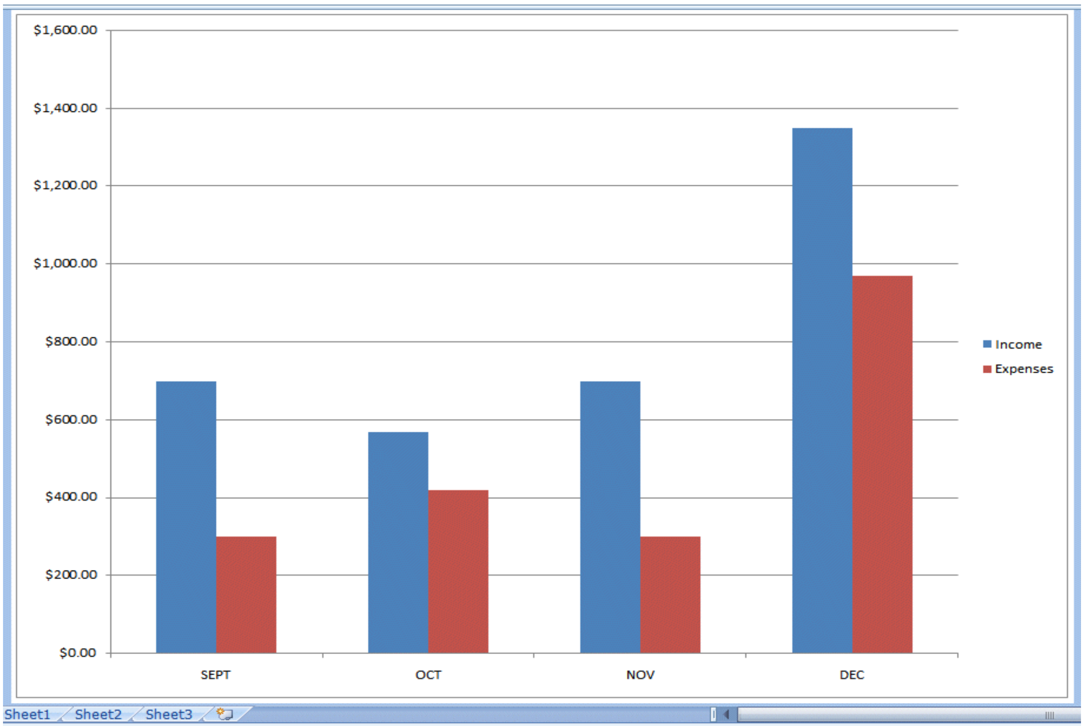


Click on the small circle to the left of **New Sheet** and change Chart 1 to **My Budget Chart** – as indicated above. Then click the **OK** button.

Look at the bottom left of your Excel screen. You will see a new Tab – **My Budget Chart**! Your data is on Sheet 1. We'll rename it when we have finished working with our chart.

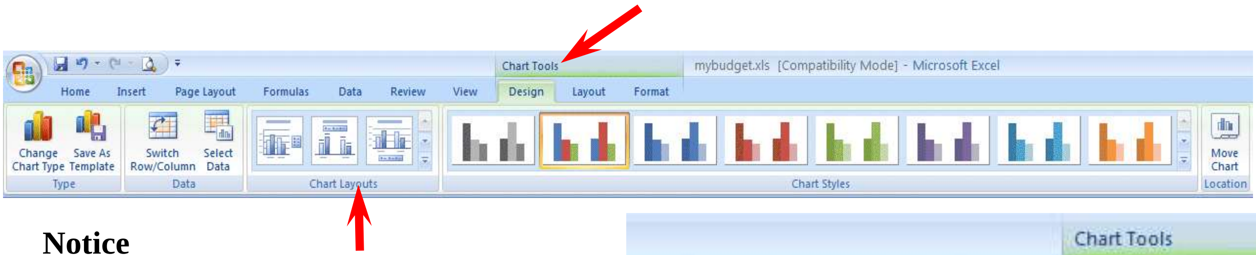


You should be on you're My Budget Chart Tab. If not, click this tab. Your chart should now fill the Excel screen. Click in one of the outside corners of your chart.

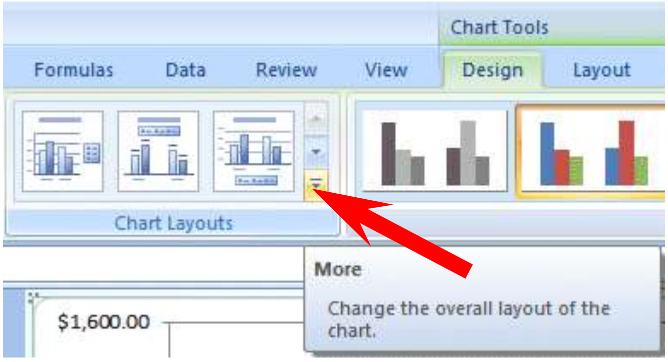


2.7.1 Chart Tools

Make sure that you can still see the **Chart Tools** Tab/Ribbon. If not, click the **Chart Tools** Tab.

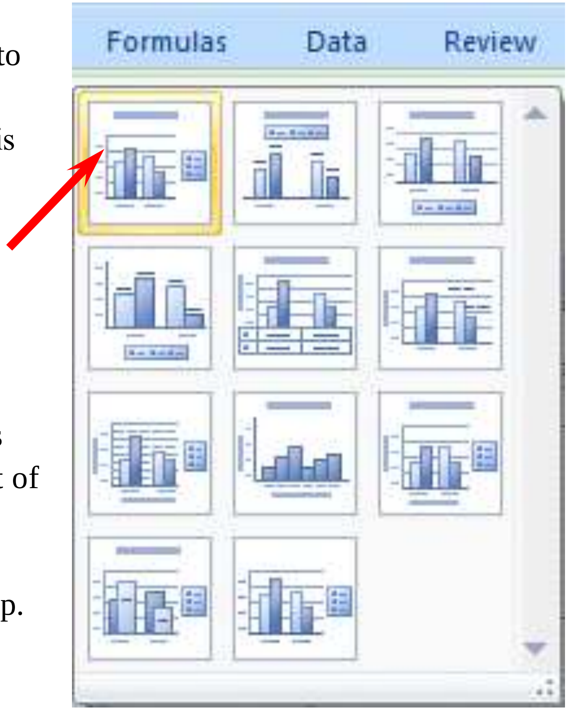


Notice
that there is a **Chart Layouts** Group in the Chart Tools Ribbon. There are many different Layouts you can choose to enhance your chart.



Click the **More** down arrow in the lower right corner of the **Chart Layouts** Group.

When you click the **more** arrow an image similar to the one on the right will appear. We'll choose the **Layout** in the upper left corner. We'll click on this choice.

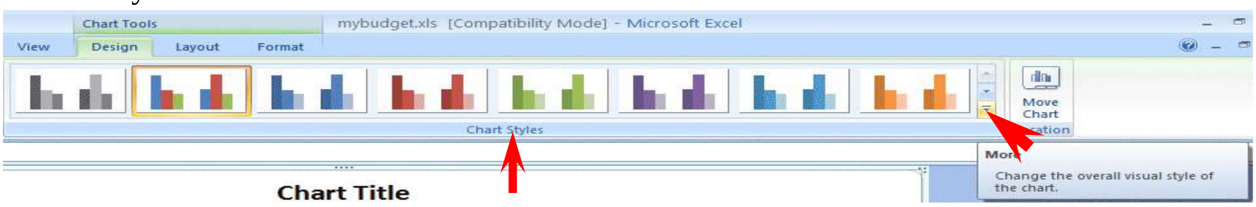


As you become more experienced with Charts, you choose the Layout that will best display your data.

You can also enhance the colors of your chart bars and backgrounds in a similar manner. To the right of the **Chart Layouts** Group is the **Chart Styles** Group. To see these **styles**, click the **More** down arrow on the lower right of the **Chart Styles** Group.

2.7.2 Chart Styles

A **Chart Styles** menu screen (*similar to the one below*) will appear. You can click various choices as you desire. Each time you click, you're My Budget Chart will change to the desired Style.

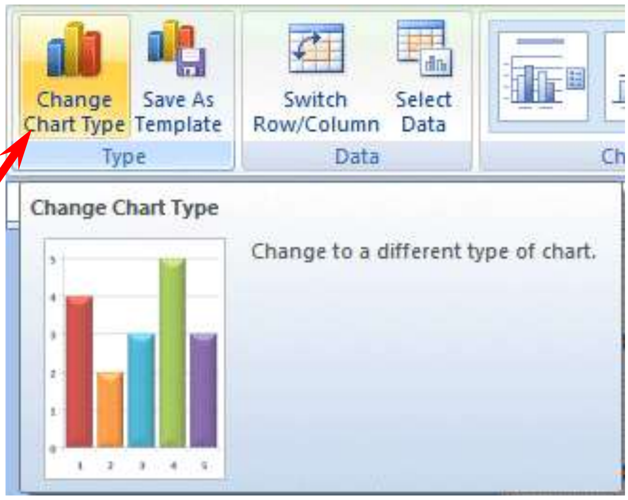


If you don't like the **Chart Styles**, you can use the **Undo** Arrow (*in the **Quick Access Toolbar***) to remove them.

Note: *As you choose different Chart Types, you will see the Chart Styles menu (above) change to that new Chart Type.*

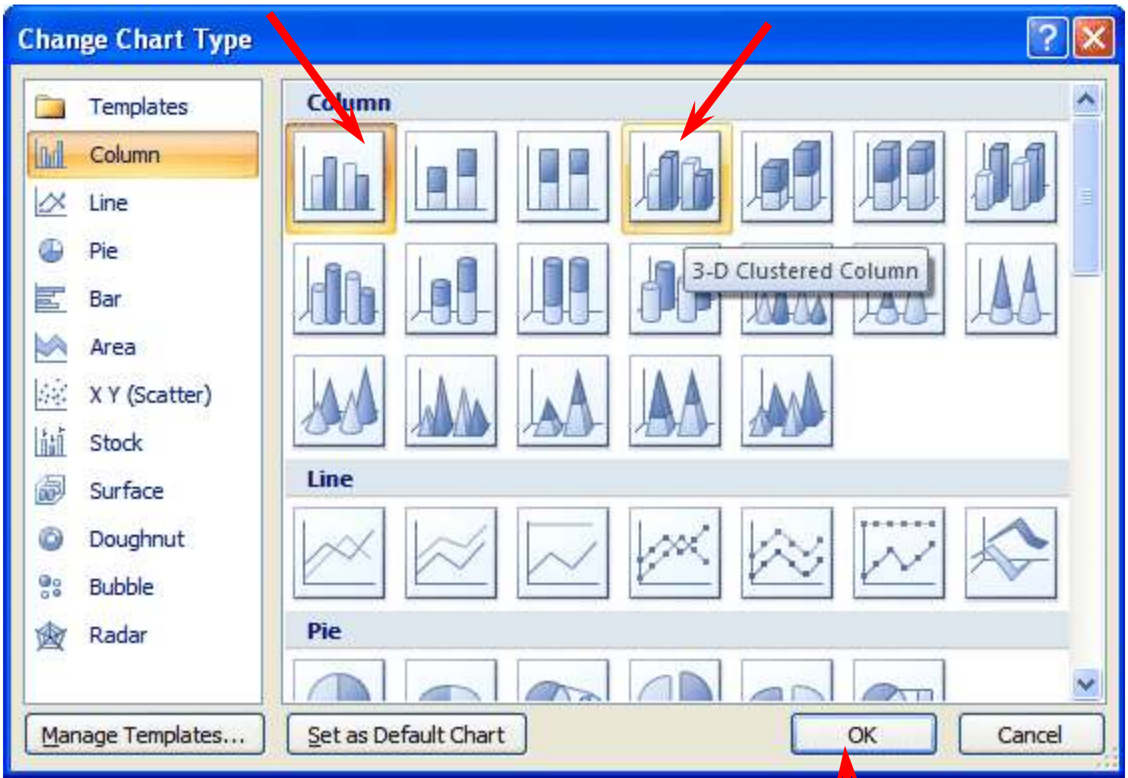
We'll show you another way to enhance the colors of your bars in a moment.

If you would like to change your Chart Type – look in the upper left corner of the **Chart Tools** Ribbon. You'll see a **Change Chart** Type button.

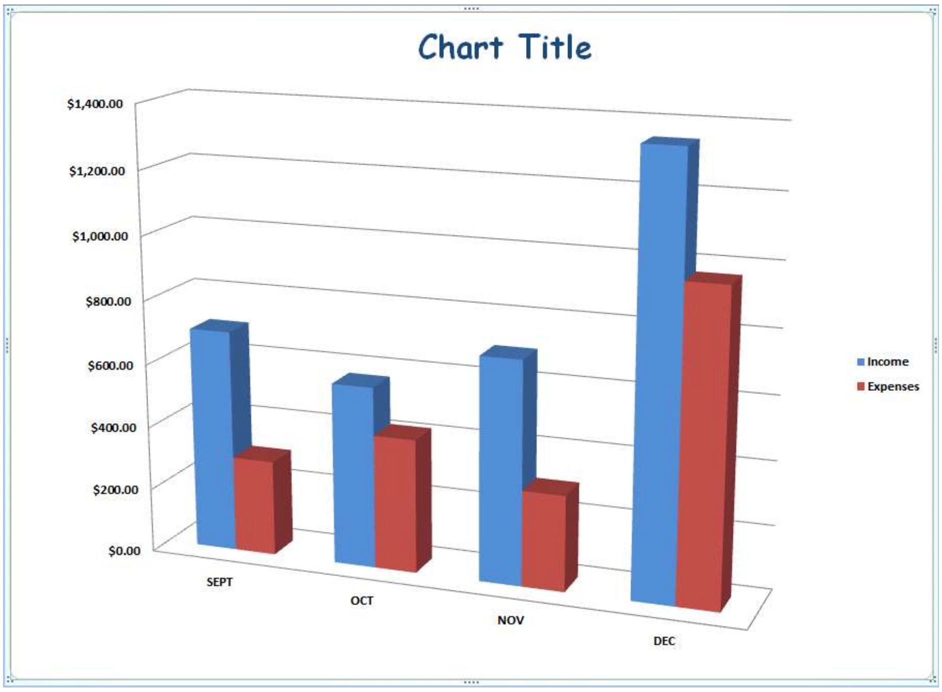


Click the **Change Chart Type button**.

A **Change Chart Type** menu screen (*below*) will appear. Currently it is on our 2-D Column Chart. We'll change it to a **3-D Clustered Column** chart next.

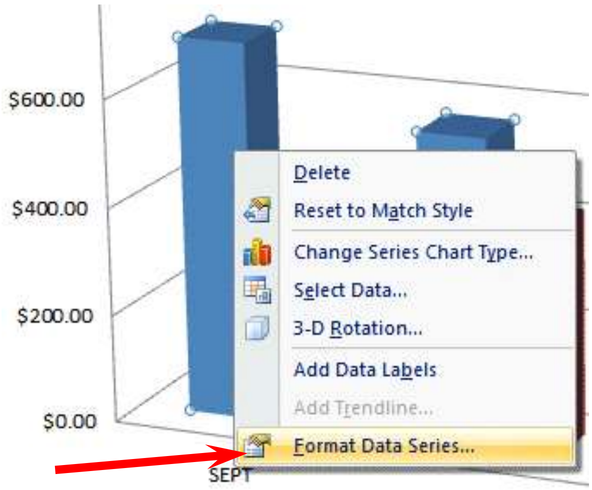


We'll click the **3-D Clustered Column** choice and then click the **OK** button. As soon as we click the OK button our entire chart changes to a **3-D Chart**.



2.7.3 Another way to change chart colors

In the previous sections, you saw how you could change the colors of the bars in your chart (***Chart Styles***). Here is another way to change colors.

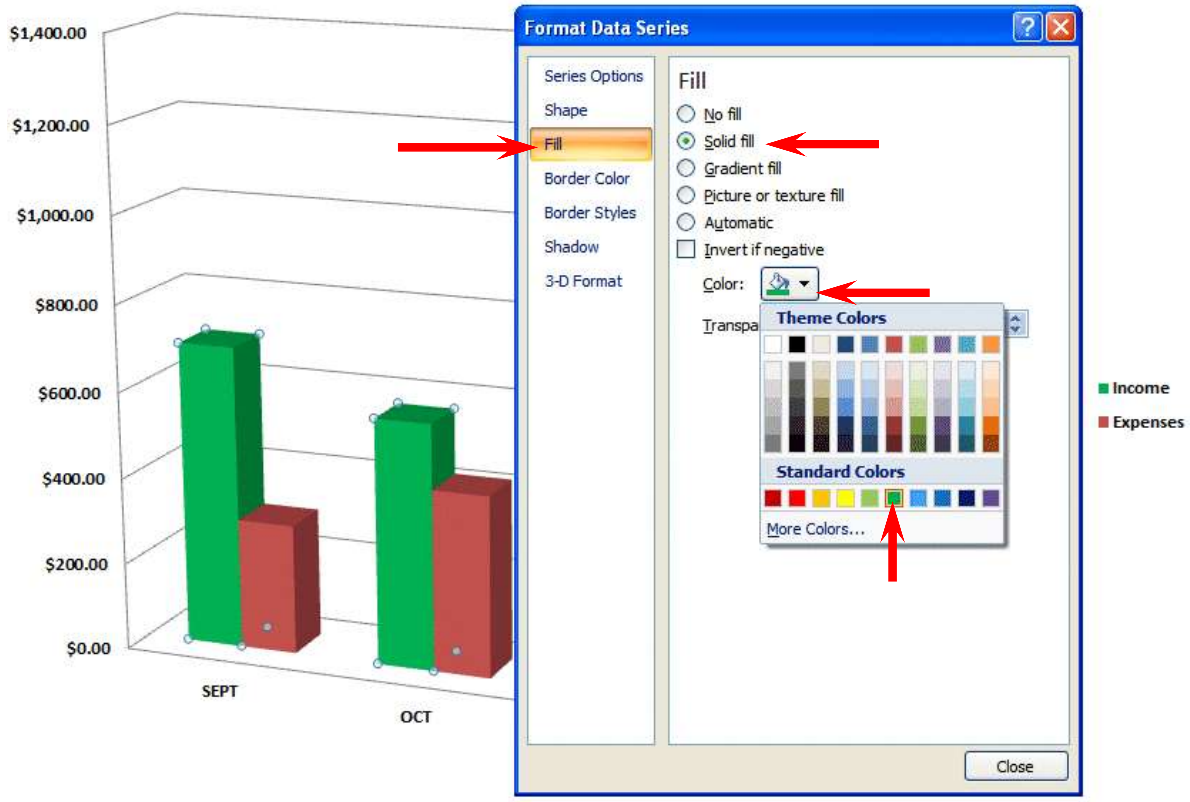


Move your cursor over one of the bars in your chart and click the right mouse button. When the pop-up menu appears, click the **Format Data Series…** selection.

A **Format Data Series** menu screen (*like the one below*) will appear over your chart. In our original chart the Income bars were blue. We think green would be a better color for our **Income** bars. So, we clicked the **Fill choice** on the left. When the Fill menu appeared on the right, we clicked the small circle to the left of **Solid fill** and then clicked the down arrow on the right side of the **Color button**. Then we clicked the **green color** choice (*if you would like to see a lot more colors you can click **More Colors…***). As soon as we clicked, our **Income bars** changed to **green**.

Notice the other choices in the **Fill menu** area. Experiment with the other choices as you desire. You can really get some amasing effects with **Gradient** and **Picture or texture** fill.

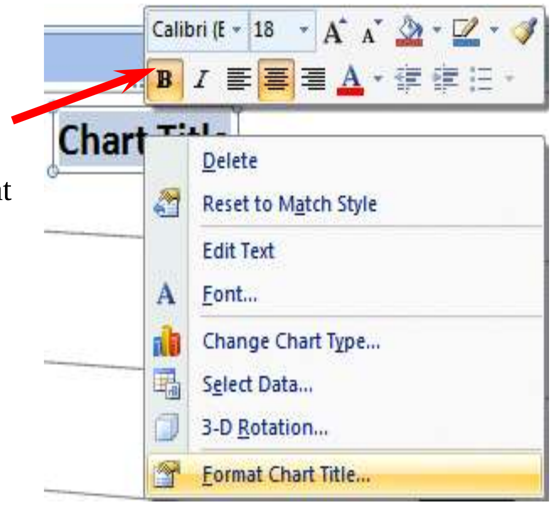
When you are **finished**, click the **Close** button.



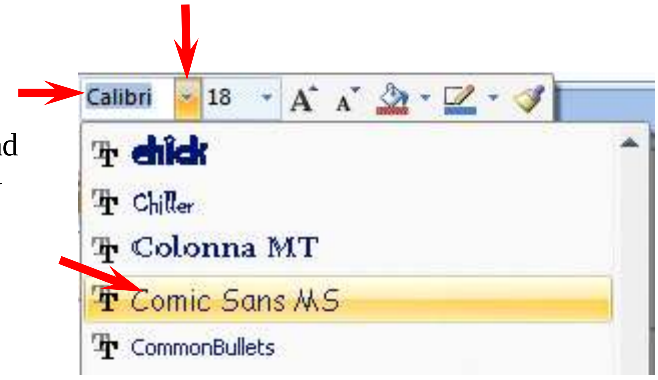
2.7.4 Changing Text

To change text in Excel 2007 you need to click on the text, the **Legend**, or the axis on which the text is located.

We right clicked on our **Chart Title** and two menus appeared. The upper menu is another good feature in Excel 2007. It's called the **Mini Toolbar**. The lower menu is the standard menu that appeared previously.



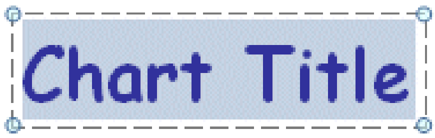
We clicked the down arrow to the right of **Calibri** and a drop down menu of font choices appeared. We moved down the list and clicked-on **Comic Sans MS**. You can equally move down the menu and choose a **font** you like.



As soon as we clicked our font choice, our Chart Title changed to that font.

You can also change the **Font size**, **Bold**, **Color**, and more, using the other *Mini Toolbar selections*. Experiment as you desire

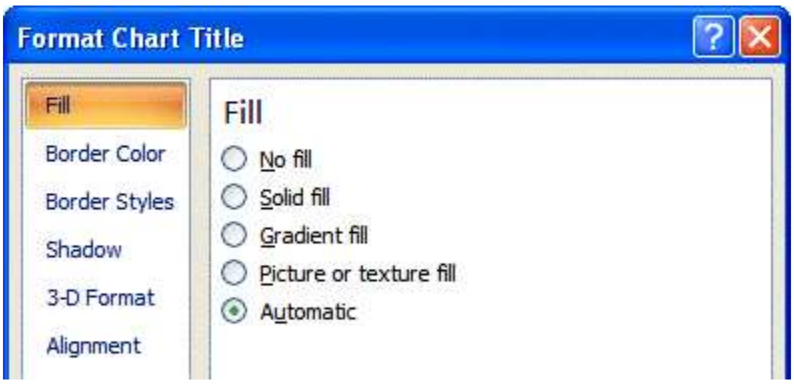
Now we'll change our **Chart Title** to something more meaningful. Move your cursor over **Chart Title** and click the left mouse button three times (*quickly*). This will highlight all the title – just like Microsoft Word.



Or, you can click and *drag* your cursor of the title to highlight Chart Title.

Type-in a title for your budget. We typed in the title you see below. When you have finished typing your title, click the left mouse button in an open area of your chart (*to turn-off and confirm the title*)

If you desire to enhance your text some more, right click on the title and the **Format Chart Title** menu screen will appear. You can use this menu to augment your text.



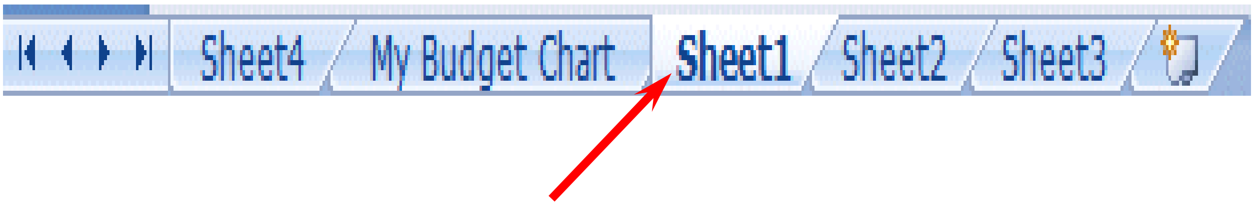
We recommend you save your work before carrying out the next exercise.

*In Excel 2007, when you save your spreadsheet***,** *you also save your graph***.** *Your graph is saved wherever you are working in the graph.*

2.7.5 Changing (Renaming) Excel 2007 Tabs

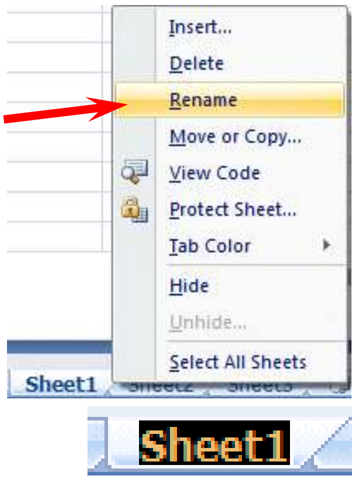
If you would like to have logical names for your Excel 2007 spreadsheet tabs rather than **Sheet 1** and **Chart 1**, we'll show you how to do this.

At the bottom of your spreadsheet you will see your tabs (*like the image on the below*). To change the name of one of the tabs, place your cursor over a tab and click the right mouse button.



We'll **change** the name of **Sheet 1** first.

Right click on **Sheet 1**and the drop down menu screen similar to the one beside will appear. Choose **Rename**.



When you choose **Rename**, the **Sheet 1 tab** will turn black (*like the image on the right***)**. As soon as you see this, type in the name you want for your chart. We typed-in **Budget Data**.

As soon as we began typing, the tab name changed. When we finished, the Tabs for our Chart and Data looked like the image below.

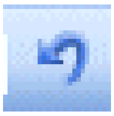
So, anytime you need to change Tab names, all you have to do is right Click and Rename.

2.7.6 Some additional Charting

Sizing your Graph Area – Click on the upper right corner of your chart area. You will see little circles now appear at the four corners and sides of the graph area. Point to the upper right corner circle and move the cursor until you see an arrow with two heads **(↔)**. Click, hold down the left mouse button and drag down to the left, then let go. Your graph will get smaller. You can make the graph area of your chart smaller or larger as you desire.

Legend Area – Click on the **Legend** on the right *(box with **Income** and **Expenses**)*. When you see the corner grabbers, make the Legend box a bit larger. Then click right in the **Legend** area. When the **Quick Toolbar appears,** make the **font bold** and **size 14**.

Printing Charts - Now let's look at your graph and then print it. Click the **Print Preview** button you placed in the **Quick Access Toolbar**. If you like what you see, go ahead and print the graph. If not, close Preview and make some more graph changes. If you have a color printer, your graph will print in color.
Whenever your graph is visible, you can point to any area of the graph, and click the right mouse button on the area, and edit that particular area. You can also click right in the chart itself.



You can go wild at this point. If you make a mistake, simply click the **undo** button at the top of the menu screen and try again.

Note: *No matter what type of printer you have, graphs take awhile to print. Be patient.*

REVISION SECTION

PAST CGCE QUESTIONS JUNE 2010 - JUNE 2017

PAPER II

GCEB JUNE 2010

1)(i) Explain Briefly the meaning of each of the following terms giving examples where necessary

(a) Automation
(b) HTML
(c) DBMS
(d) Data Flow Diagrams (DFD)
(e) Data Protection (15marks)

ii) How does the Data Protection Act define the following terms?

(a) Data
(b) Data subject
(c) Personal data. (5 marks)

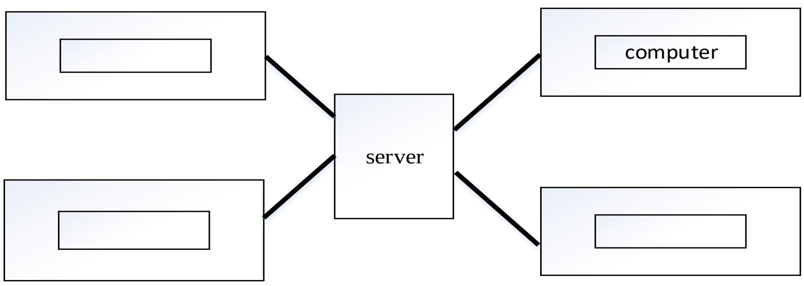
---

2) i) a) What is teleworking? (2 marks)

b) List two advantages of teleworking to a teacher. (2 marks)

c) List and describe two advantages of teleworking to a school. (4 marks)

ii) Below is a diagram of the structure of a typical network topology



(a) Give the name of the network topology above. Give the name of another typical network topology. (2 marks)

(b) Give two reasons why it is better for students to use networked computers. (2 marks)

(c) List two functions of a file server on the network. (2 marks)

(d) The school has an intranet which students can log on from home through a WAN.

(i) Explain the meaning of WAN. (2 marks)

(ii) List two different ways in which the intranet can help students in their studies. (2 marks)

---

3)i) a) What is ergonomics? (2 marks)

b) List and explain three physical health matters related to the use of computers. Give possible ways of avoiding each of these problems. (6 marks)

ii) a) Explain the use and functioning of the following
- Bar code readers
- Optical Mark Recognition (OMR). (6 marks)

b) List two main functions of each of the following hardware
- Video card
- Sound card
- Network Interface Card

---

4)i) (a) Explain the meaning of the term data. (2 marks)
(b) List and explain two main threats to the safety of data and for each give two measures which are used to fight against it. (6 marks)

ii) (a) Explain how each of the following methods could be used to implement a system

- Parallel running
- Direct changeover
- Phased implementation. (3 marks)

(b) Give one advantage and one disadvantage of each of the methods in (ii)a. (6 marks)

---

5) Read the text below and answer the questions that follow:

APPLICATION CONTROL STANDARDS (ACS)

ACS includes policies and procedures associated with user activities and the automated controls designed into applications. Controls could be in place to address both batch and on-line environments.

ACS enhances the security, integrity, and reliability of automated systems by ensuring that input, processed and output information are authorized, accurate, complete and secure. Management considers these issues at the outset of a project and includes them as soon as possible in applications and systems design. Controls are usually categorized as preventive, detective or corrective. Preventive controls are designed to prevent unauthorized or invalid data entries. Detective controls help identify unauthorized or invalid entries. Corrective controls assist in recovering from unwanted occurrences.

i) What do you understand by batch and on-line environment as used in the text. (4 marks)

ii) Give the purpose of automated input controls in (i). (3 marks)

iii) Group the following control methods in preventive, detective and corrective controls.
Check digits Completeness checks

Range checks   Duplication checks
Format checks   Error reporting
Limit checks   Validity checks   (8 marks)

iv) Explain how you will implement formal and range checks in a spreadsheet. Use examples from: Name, Sex, Date and Marks fields. (5 marks)

---

6)i) (a) Explain the meaning of the term ''expert system''. (4 marks)
Give two situations where expert systems are used other than in playing the game of chess. (b) Describe how a user interacts with an expert system. (3 marks)
ii) (a) Name and describe two types of electronic conferencing techniques giving the special hardware required for each type. (6 marks)
(b) Give two benefits to an organization and its employees of using electronic conferencing techniques. (2 marks)
iii) State three advantages and three disadvantages of using e-mail. (5 marks)

---

7)(i) Name four computer crimes in Cameroon and briefly describe each of them. (4 marks)

(ii)For any two of the crimes you state for each, two ways that you can use to prevent them(4marks)

(iii)State and explain two reasons why a data protection law is necessary in Cameroon. (4marks)

iv) Briefly describe what the following computer professionals do:
(a) Database administrator
(b) Systems analyst
(c) Computer programmer
(d) Networks system administrator. (8 marks)

---

8)i) High level languages are machine independent while low level languages are machine oriented

a) Give one example of a high level language. (1 mark)
b) Explain the meaning of the following two terms:
- Machine independent
- Machine oriented. (3 marks)

c) Low level and high level languages are different. List three features of each language level that is not found in the other. (6 marks)

ii) In a given firm, the wages of workers are calculated as follows:

- Wages depend on hours worked and pay rate (amount per hour)

- Wages for extra hours greater than 40 have rate of 1.5 times per rate.

a) Write and algorithm that receives the number of hours worked and the pay rate as input and the calculates the wage of a worker. (7 marks)

b) Use your algorithm to calculate the wage of a worker who worked for 43 hours at a pay rate of 1,000frs. (3 marks)

---

9)i) Distinguish between network topology and network configuration. (2 marks)

ii) Name and describe two types of network configurations. (4 marks)

iii) You were asked to design computer networks in different geographical areas. Choose what type of network you will design for the ones below.

a) A network to link a laptop, a desktop, a digital camera and a PDA in a corner of a room

b) A network to link all the networks of a multinational company' s offices in two continents.

c) A network to link computers of a company' s offices in one large city.

d) A network to link all computers in many buildings of a college inside a fence. (4 marks)

iv) Convert from base 2 to decimal the number 101101.01011. Convert the decimal number 268 to octal. Convert the binary number 11101011011 to octal. (4 marks)

v) Design and complete the truth table of the following logic operators NOR and AND. (2 marks)

vi) Define an Operating System (OS). Give two functions of an Operating System. (4 marks).

GCEB JUNE 2011

1. People who work in front of a computer monitor for long hours usually expose themselves to some health hazards.

(i) Name four health hazards that could directly be linked to prolonged use of computers. (4 marks)

(ii) List and explain four actions you would take to avoid the four different computer related hazards (8 marks)

(iii) a) What is a robot? (2 marks)

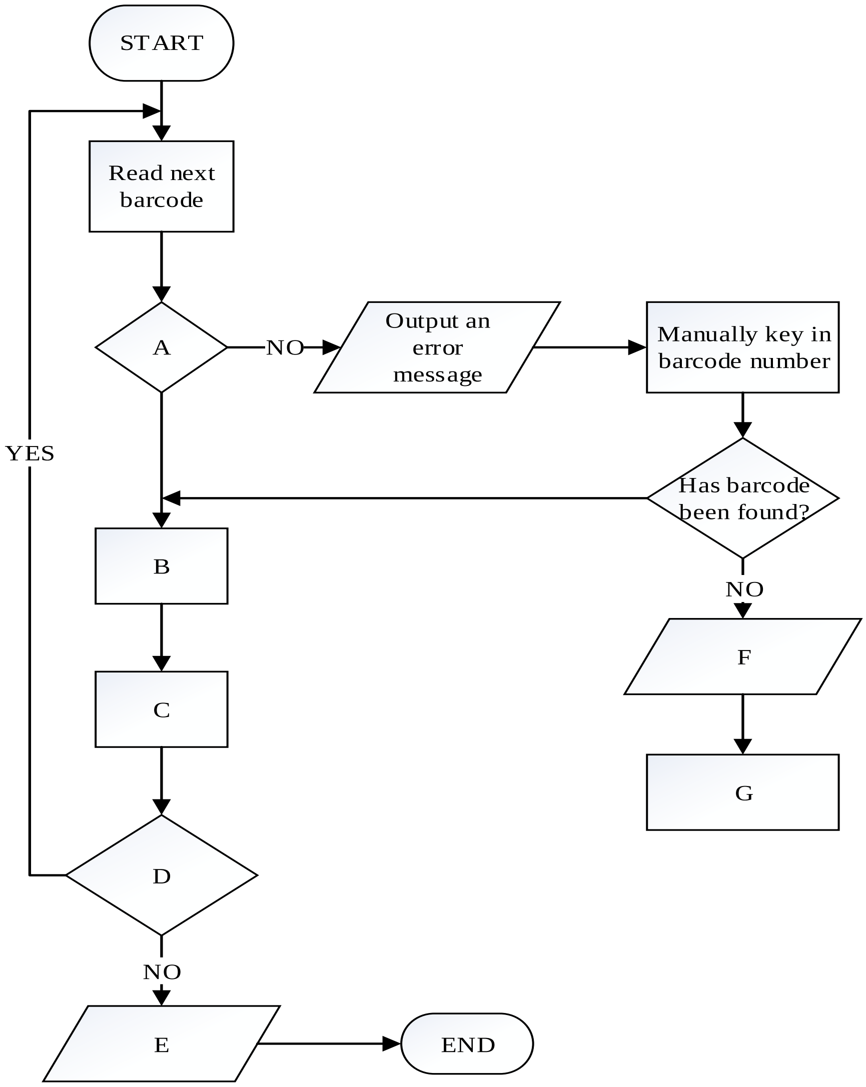
(b) State three advantages of robots over human workers. (3 marks)

---

2. (i) The following flow chart show how the barcode written on the item is used to find the price, do stock control and produce an itemized bill. Select statements from the list below to match the letters A to G in the flowchart. (You may not draw the flowchart).

List of Statements

1. Any more barcodes to read?
2. Has barcode been found in the file?
3. Look up the price of the item in the file?
4. Output error report.
5. Output itemized bill.
6. Reject item.
7. Update stock file.



(7 marks)

(i) An analyst is hired to carry out a feasibility study.

a) List and describe three areas that the system analyst should report on in the feasibility study. (6 marks)

b) Describe the methods that the system analyst could use in order to collect information about the current being used in the organization (4 marks)

3.(i) Describe the following, stating an example where applicable.

a) Expert systems.
b) Geographical information systems.
c) Simulation systems.
d) Reservation systems (12 marks)

(ii) a) Give two implications of e-commerce to society. (2 marks)

(b) List three reasons why e-commerce may not be a good business strategy in Cameroon. (3 marks)

---

4. On line banking has improved banking in most developed countries. It allows customers to access their accounts 24 hours a day without the need to open the door of branches. The central computer runs most of the banking processes.

i) What do the following terms mean?
   a) Online
   b) Central computer (4 marks)

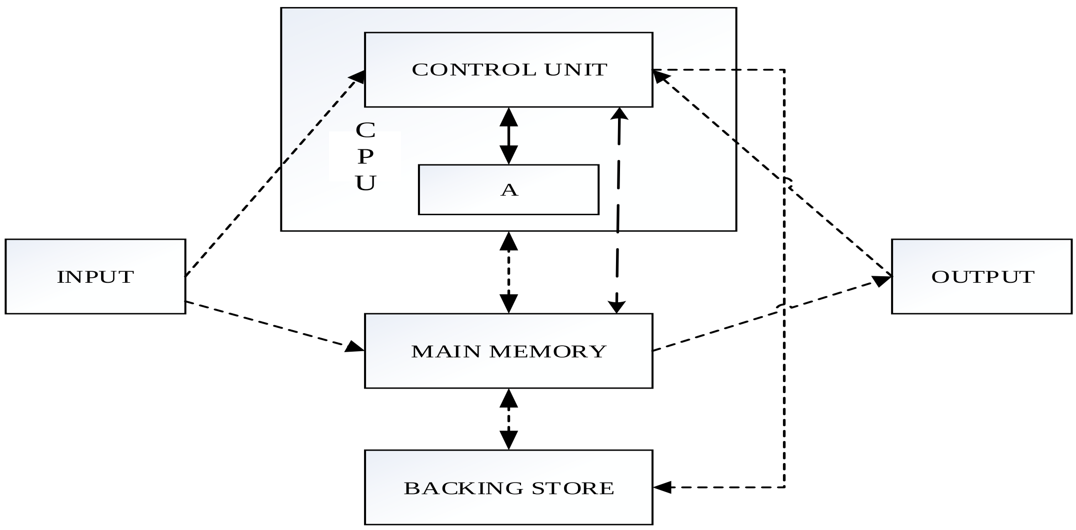
ii) Give on advantage and disadvantage of online banking on the
   a) Bank
   b) Customer (4 marks)

iii) Briefly state and describe the main function of each of the following computer hardware components:
   a) CPU
   b) RAM (6 marks)
   c) Motherboard

iv) Draw the truth table for the following Boolean expressions:
   X and (X OR Y) (3 marks)

---

5. i) The diagram below shows the logical structure of a computer system. It shows the CPU, the main memory, backing store input and output.



a) Give the name of the component labeled A and state its main function. (3 marks)
b) Give another name for the main memory and state its major function. (3 marks)
c) How the seize of the main memory influence the computers memory. (3 marks)
d) Give the components found in the CPU that are not shown in the diagram (2 marks)

(ii) Complete the table by stating the basic principle by which the given printers operate (4marks)

Ink jet	Sprays ink through tiny nozzels on the paper
Laser	
Dot matrix	

(iii) State two advantages of matric printers (2 marks)

6.(i) What is the main difference between a floppy disk and a compact disk taking into consideration method of storage (3 marks)

(ii) a) What is the storage capacity of standard $3\frac{1}{2}$ floppy disk? (1 mark)

b) How many floppy disks can contain the same amount of information in a standard 750 MB CD? (3 marks)

(iii) Define the following disc terminologies

a) Tracks
b) Sectors (4 marks)

iv) Why is data storage on a hard disk organised in cylinders? (2 marks)
v) Why should a hard disk be defragmented regularly? (2 marks)
vi) Why is it necessary to format a magnetic disk before it can be used? (2 marks)

---

7. (i) In a school, teacher-student after school communication is by wireless network of laptop computers.

a) State two advantages for teachers using this system rather than desktop computers located in a computer laboratory. (4 marks)
b) Give two disadvantages of using lap top computers rather than desktop computers. (2 marks)

(ii) A university provides online training courses.

a) Give two uses of each of the following to help the university run these courses

- Spreadsheet package
- Database package

- Desktop publishing (6 marks)

b) A leaflet designed using a word processing software to advertise a course is currently too big to fit on a single printed page. State how you will use three features of word processing software to alter the design so that it can fit on one page. (3 marks)

c) Give two disadvantages of a database package over a spreadsheet to manage school information. (2 marks)

8.(i) a) What is SDLC? (1 marks)
b) List 4 phases in the SDLC. (2 marks)
c) Choose two phrases from (b) above and state 2 activities for each phase. (4 marks)
d) Describe two methods by which we can implement a completed computer system. (4 marks)
(ii) Explain each of the following project management technology
a) Project
b) Critical path
c) Task (6 marks)

9.(i) Define computer network. (2 marks)
(ii) Distinguish between computer network topology and computer network configuration. (2 marks)
(iii) Name and describe one network configuration. (3 marks)
(iv) With the help of annotated diagrams, name and describe two computer network topologies. (4 marks)
(v) Two types of network connecting media are cables and wireless standards. Name two types of network cables and two wireless network standards. (4 marks)
(vi) Give the main difference between a hub and a switch. (2 marks)

GCEB JUNE 2012

1.(i) Expand the following acronyms giving explanations in each case

a) BCD
b) ASCII
c) EBCDIC (3 marks)

(ii) What is a robot? (1 mark)

(iii) Explain two uses of computers in the following organizations:
a) Military
b) Education
c) Hospital
d) Multinational corporation
e) Manufacturing industry (10 marks)

(iv) a) Convert the decimal number 120 to an octal number
(b) Convert the binary number 101101001 to an octal number (4 marks)

(v) Design and complete the truth tables of the following logic operators OR and NAND. (2marks)

---

2. (i) Give three reasons why data misuse is more easily carried out in a computer system than in a manual system. (3 marks)

(ii) (a) What is a computer virus? (2 marks)

(b) State two actions of a computer that may suggest the presence of any virus in it. (2 marks)

(c) Write down four ways in which you can prevent a computer virus infecting your computer. (4 marks)

(iii) What aspect of computer security does each of the following describe?

(a) Prevent access to unauthorised computers(persons).
(b) Detect keystrokes logging or unauthorised data transmission.
(c) Used to identify the user to a computer system.
(d) Used for security and changed regularly.
(e) Restrict access to certain users.
(f) Data may be stolen but cannot be read. (6 marks)

(v) State and briefly describe one other security method to protect the data of an organization. (3 marks)

3(i) (a) Name and briefly describe three internal threats to ICT systems. (6 marks)

(b) Name and briefly describe two external threats of an ICT system. (4 marks)

(c) Describe three measures you will take to protect your ICT system from illegal access. (3 marks)

(ii) (a) Explain what is meant by:

- File name
- File compression
- File Allocation Table (FAT). (3 marks)

(b) Give two reasons why it is necessary to defragment a hard disk. (2 marks)

(c) The file compression ratio of a disk compression utility is set at 15:1. If the content of the hard disk is28 GB before compression, what will it be after compressing the data with this compression utility? Give your answers in GB and MB. (2 marks)

4(i) (a) What is batch processing? (2 marks)

(b) How is a batch processing system different from real-time processing system. (1 mark)

(ii) (a) Which two of the following systems are batch processing systems

- Reservation systems
- Payroll
- Billings

Explain your choice of answers. (4 marks)

(b) Give three ways in which office staff will be affected by introduction of a computer system. (3 marks)

(iii) (a) Explain the following terms: Modelling and simulation. (5 marks)
(b) Describe situation which can be modelled and simulated. (5 marks)

---

5. (i) Computer users with different disabilities require special hardware and software. Complete the table below stating in each case, special hardware and/or software for people with the given disabilities.
(5 marks)

Disabilities	**Hardware/software**
Limited mobility	
Armless	
Slow typist	
Blindness	

(ii) (a) What is data capture? (2 marks)
(b) Give three examples of automatic data capture methods, explaining how each method is carried out.
(4 marks)
(iii) (a) What do you understand by intranet and extranet. (3 marks)
(b) List three special features of the networks mentioned in (a) above. (3 marks)
(c) How useful are these networks to an organisation? (3 marks)

6.(i) Define each of the following terms
a) Telecommuting
b) Videoconferencing
c) Expert system
d) Loader (4 marks)

(ii) What is system software? (2 marks)
Name and describe four categories of system software and give an example for any two. (6 marks)
(iii) Fully expand each of the following acronyms giving a brief explanation of its use in each case
a) GIS
b) HIS
c) EIS
d) LIS (4 marks)

(iii) What is Management Information System (MIS). (1 mark)

(iv) Reports are very essential for managers who use them to make good decisions. Name and briefly explain three types of MIS reports. (3 marks)

7. A payroll system has just been created for an agricultural fir, of 50 workers.

   i) Describe three main steps that may be taken to implement the system. (3 marks)
   ii) Describe two verification methods that the firm could use when entering data into the system. (4 marks)
   iii) Describe three system input controls using specific examples. (6 marks)
   iv) Describe two measures taken by the organization to ensure that data is not accidentally lost. (2 marks)
   v) Describe how, upon installation, the system can be tested. (5 marks)

8. A school has a manual library system which is run using the following rules:

   - Books are grouped according to subjects in shelves.
   - Books have codes on them.
   - Students have access to the library.
   - Each student has a school ID card.
   - The record of books borrowed by students is stored on cards which are kept in the library.

   i) Give six data structures (fields and data types) of the details found in a borrower's card. Present your answer in a tabular form. (6 marks)

   ii) State and explain three ways of collecting information about the current system. (6 marks)

   iii) State four items of information that should be part of the design of the computer system. (4 marks)

   iv) Design a suitable output format for displaying the details of a borrower. You may annotate your design to make it clear. (4 marks)

9. The determinant (det) of a 2x2 matrix

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

Is given by det = ad - bc, where $a{\cdot}d \equiv a{\times}d$

The table below shows the values of a 2x2 matrix entered in a spreadsheet

	A	B	C
1	4	6	
2	5	9	
3			
4			
5			
6			
7			
8			

Write down the formula you will use to calculate the determinant of the matrix whose entries are in the table. (2 marks)

i) Rewrite the equivalence of the formula above using a spreadsheet function, such as MDETERM in excel, which computes the determinant for any m *x* n array, or otherwise. (2 marks)

ii) Write a program in a stated programming language that reveals inputs of four integers, a, b, c and d, and displays the determinant of the matrix obtained from them as given above. (8 marks)

iii) (a) What is pseudocode? Give two rules for writing pseudocodes. (4 marks)
(b) What is the difference between pseudocodes and source codes? (2 marks)
(c) Give the importance of pseudocodes. (2 marks)

GCEB JUNE 2013

1. (i) Explain, using examples, the following computer terms.
   (a) Data verification
   (b) Simulation
   (c) Handshaking
   (d) video conferencing
   (e) Batch processing (10 marks)

(ii) (a) Give two examples of computer crimes. (2 marks)
(b) Give two methods of preventing the state crimes in (a) above (2 marks)

(iii) E-commerce is carried out online. State three possible effects on society if e-commerce increases greatly. (3 marks)

---

2. (i) State three (3) tasks to perform in the design stage of the development of a new computer system. (3 marks)

(ii) A national bank has changed from a manual filling system to a computer-based system.
(a) State two benefits to the bank for the change. (2 marks)

(b) State two effects on the staff due to the introduction of a computer-based system(2marks)
(c) Give two reasons why the bank would prefer a parallel conversion to the new system (2 marks)

(d) Give two examples where parallel conversion would not be suitable (2 marks)

(iii) Virtual reality is a 3-Dimensional world created by the computer in which the user is part of.

(a) Give two examples of the special hardware needed to interact in a virtual reality. (2 marks)

(b) Give two advantages of using virtual reality (2 marks)

(c) Give an example of an application which uses virtual reality. (2marks

---

3. (i) There are two different kinds of graphics stored in a computer. Bit-mapped and vector-based graphics. Explain the meaning of bit-mapped graphics and vector-based graphics, giving one file format (extension) used for each. (6 marks)

(ii) Give three advantages of vector-based graphics over bit-mapped graphics. (3 marks)

(iii) Differentiate between data integrity and data security. (2 marks)

(iv) Name and explain three measures you should use to ensure data integrity and three measures to ensure data security. (6 marks)

---

4. (i) What is meant by pseudocode?

(ii) Explain the following algorithm development concepts:

(a) Stepwise refinement

(b) Top-down design (4 marks)

(iii) Briefly define and say what each of the following programming tools are used for:

(a) Compiler

(b) Interpreter (2 marks)

(iv) Give the structure for the following loop constructs. Your structure should indicate the loop (or test) condition and the loop body (or statement). (The structure of a choice construct is shown below as an example). Indicate for (a) and for (b) the loops condition under which the loop would continue to iterate.

(a) Repeat-until (or do-while)

(b) While

(c) For (3 marks)

Example:

```
IF condition THEN
   Statement
ELSE
   Statement
```

---

5. (i) (a) Draw a block diagram that shows how input, process, main memory, output and storage relate in a computer. (5 marks)

(b) Select any **TWO** blocks of your choice from (a) and name a device for each (2marks)

(ii) Using the block diagram, give the definition of a computer (2 marks)

(iii) Name the two blocks involved in the machine cycle. (2 marks)
(iv) State the four stages of the machine cycle (4 marks)
(v) Explain the term volatile memory and give an example. (2 marks)

---

6. (i) Describe the main difference between applications software and systems software (4marks)
(ii) Name four types software and give one use for each class. (4 marks)
(iii) Name the type of application software you would use for the following purposes.
(a) To create, edit and format text based documents.
(b) To work with numbers, calculators, charts and graphs.
(c) To organise and access large amounts of information
(d) To produce good news letters and company magazines (4 marks)
(iv) (a) Define the term computer virus and computer anti-virus. (2 marks)
(b) Briefly explain how the following viruses work:
- Trojan horse
- Worm
- Boot sector virus. (3 marks)

---

7. (i) Give TWO uses of each of the following information systems. Your answer should include the type of data or information they manage, and also state how useful the information is to the concerned organisation
(a) Patient record system
(b) Payroll system (6 marks)
(ii) State and briefly explain TWO things that determine the design of interfaces for different groups (students, staff, administrators, etc) in a school. (6 marks)
(iii) (a) Convert 4301_8 to binary.
(b) Evaluate 101_2 x 11_2 (5 marks)

---

8. (i) (a) Define the term system development life cycle. (2 marks)
(b) List four stages of the SDLC. (4 marks)
(c) For any two of the stages given in (b) above briefly describe the major tasks involved. (4marks)
(ii) (a) What is a programming language?
(b) Give one difference between low level languages and a high level language? (4 marks)
(iii) With respect to programming, explain the following:
(a) Portability
(b) Algorithm
(c) Control Structure. (3 marks)

---

9. (i) (a) what is internetworking?
Define the term "The internet" (4 marks)
(b) Describe the terms:
- Intranet
- Extranet. (2 marks)

(c) What is a search engine? State one use of search engines. (3 marks)

(ii) Explain what each of the following hardware is used for in a computer network:

(a) Switch (2 marks)

(b) Repeater (2 marks)

(c) Router (2 marks)

(d) Multiplexer and Demultiplexer (2 marks)

GCEB JUNE 2014

1. (i) Research on Automatic Speech Recognition (ASR) has intensified since the beginning of the 21st century. Emerging technologies are on the way to increase the quality of interaction between the users and the machine.

(a) What is automated speech recognition? (2 marks)

(b) State two advantages and limitations of speech recognition. (3 marks)

(ii) Explain briefly the use of the following input technologies and site a situation in which each can be useful.

(a) Magnetic STRIP (2 marks)

(b) Magnetic ink character recognition. (2 marks)

(c) Touchpad (2 marks)

(iii) (a) Difference between data validation and data verification (2 marks)

(b) Give the meaning of ***proof reading*** and ***double entry*** (2 marks)

(c) Describe a situation or application that use the double entry method (2 marks)

---

2. (i) Give the full meaning of the following and state an application for each:

(a) LIFO (2 marks)

(b) FIFO (2 marks)

(ii) (a) What is a source code (2 marks)

(b) Give a reason why programming languages are classified as high level languages? (1 mark)

(c) State two ways in which you can represent an algorithm (2 marks)

(d) Describe one way in which algorithms can be compared. (2 marks)

(iii) (a) What artificial intelligence in an industry (2 marks) (b) What is Computer Aided Design? (4 marks)

Name one professional who uses CAD and describe to what extent it is useful to him. (4 marks)

---

3) (i) Define the following terms:

(a) Relation Database. (2 marks)

(b) Database management systems. (2 marks)

(c) Redundancy in database. (2 marks)

(ii) Give advantages and disadvantages of a DBMS. (5 marks)

(iii) Propose two domains of application of each of the following systems:

(a) Geographic Information System. (2 marks)

(b) Reservation System. (2 marks)

(c) Expert System. (2 marks)

---

4) (i) Give the main function of the following hardware in a computer system:
(a) Video Card (2 marks)
(b) Ram chip (2 marks)
(c) Ethernet Card (2 marks)
(ii) Define the following terms:
(a) Simulation (2 marks)
(b) Videoconferencing (2 marks)
(c) File compression. (2 marks)
iii) (a) What is ergonomics? (2 marks)
Give two advices you will give to a typist who is suffering from back pain and eye strain. (2marks)
(b) Give two advantages of LCD monitors over CRT monitors. (2 marks)

---

5) i) A machine cycle consist of a series of operations performed to run a single program instruction fetch, decode and execute the instruction and return the result to the RAM if necessary.
a) What is the meaning of RAM? Give the main function of the RAM (3 marks)
b) Explain the process of fetch decode and execute in a machine cycle. (3 marks)
(ii) Describe the following
(a) Word size
(b) Byte
(c) ASCII code (6 marks)
(iii) Name five components of the CPU and briefly explain the function of each. (5 marks)

---

6. (i) Define the following web technologies and give an example of each.
(a) Web page
(b) Search Engines
(c) Browser (6 marks)
(ii) (a) give three impacts of computer virus to an educational institution.
(b) Suggest two methods that can be taken by a school secretary to protect the computer from being infected by computer viruses.
(c) Give two effects of a virus to a computer system. (7 marks)
iii) a) What do you understand by the term multimedia? (1 mark)
b) Give three ways in which the installation of multimedia centers in Cameroon would affect both teachers and students? (3 marks)

---

7. (i) Define the following terms
(a) Computer Ethics
(b) Hacking
(c) Copyright (3 marks)
(ii) Give the difference between
(a) Copy-paste action and cut-paste action
(b) Word art and Clip art (4 marks)

(iii) Complete the table below by stating the most appropriate type pf software program that can be used for the stated tasks. (4 marks)

	TASK	Type of software
a	Conversion of data to charts	
b	Calculation of areas of a figure in which the user supplies the required input and software generates the output.	
c	To present information on the activities of an organization to the board of directors during a seminar.	
d	Create, manipulate and format text, images and tables for publicity.	

(iv) Give the difference between the different pair of transition modes:
(a) Parallel and serial
(b) Simplex and full-duplex
(c) Synchronous and asynchronous (6 marks)

---

8. (i) (a) What is Pseudocode?
Give importance of Pseudocode in programming (3 marks)
(b) Give three qualities of a good Algorithm (3 marks)
(ii) (a) What is fault tolerance? (2 marks)
(b) What device can be used to amplify signal to cover a long distance? (1 mark)
(c) Complete the table below by giving the reason for each of the aspects of network topology.

	Aspect	Reason
1	Signal coalition is not common in a ring	
2	Mesh offers the greatest fault tolerances	
3	Security is more assured in a star than in a bus.	
4	A ring offers more fault tolerances than star	

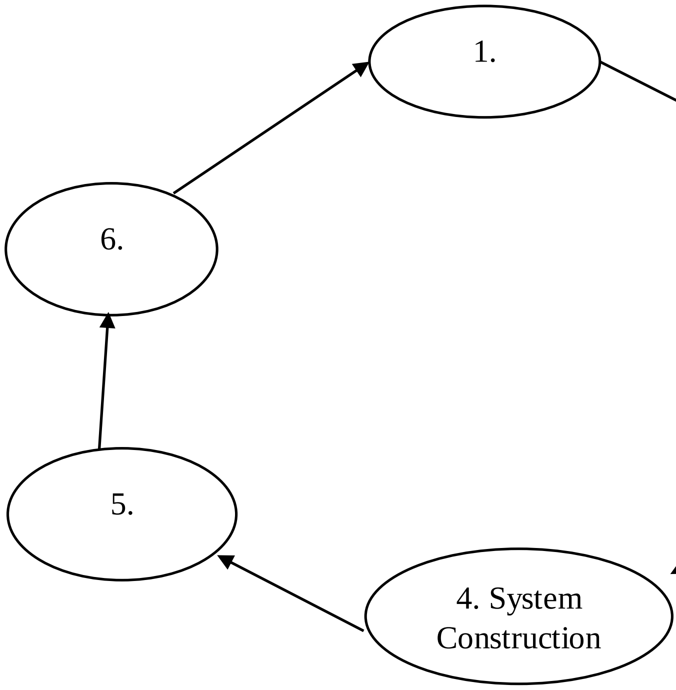
(4 marks)

(iii) Give the full meaning of each of the following file formats and state the purpose for which they are used
(a) .PDF
(b) .MPEG (4 marks)

---

9. (i) Evaluate the following, where binary numbers are absolute values:
(a) 34_{10} to Binary
(b) $10110101_2 - 11111_2$
(c) $101101_2 \times 101_2$ (4 marks)

(ii)



(a) Sketch the diagram above and insert the missing phases (4 marks)

(b) In the analysis of the SDLC, different tools are available to assist the system analyst and end user. Name and explain any two of the tools used. (4 marks)

(iii) Give two reasons why:

(a) The operating system is often installed before any other software.

(b) People prefer working with graphical user interface than command line interface. (4marks)

GCEB JUNE 2015

1. (i) (a) Explain two reasons why robots are useful to industries. (2marks)

(b) Describe two reasons that can be taken to secure a banking system (4marks)

(ii) (a) Describe a DBMS. (2marks)

(b) What is relational database? (2marks)

(c) Give two examples of a flat file in databases. (2marks)

(iii) (a) Explain the term "Programming paradigm". (1mark)

(b) Define the following programming paradigms

- Procedural programming
- Declarative programming (4marks)

---

2. (i) (a) What is network topology? (2marks)

(b) Explain two considerations in selecting a topology for a school. (4marks)

(ii) (a) What is prototyping? (2marks)

(b) Give two reasons for prototyping. (2marks)

(iii) (a) What is a program control structure? (2marks)

(b) Using either Pascal or C programming language, write out the general form for the following control structures.

- Sequence (1mark)
- FOR statement and (1mark)
- WHILE statement (1mark)

(c) Draw the flowchart symbols for the choice/selection structure. (2marks)

3.(i) (a) Write out the steps involved in the SDLC in an annotated diagram. (2marks)

(b) Distinguish between piecemeal and direct system implementation methods. (2marks)

(c) Describe one other alternative method of building a system. (2marks)

(ii) (a) State two functions of a project manager in project management. (2marks)

(b) With respect to project management, describe a PERT chart, stating all the elements involved. (2marks)

(c) That is the importance of identifying the critical path of a project. (1mark)

(iii) Discuss TWO possible effects on people who live in some developing countries of having no access or very limited access to ICT. (6marks)

---

4. (i) (a) What is a file format? Give one type of file format. (2marks)

(b) Explain one importance of file format. (1mark)

(c) State two reasons why file formats vary. (2marks)

(ii) (a) Systems need protection from hazards e.g natural hazards such as floods and fire, Describe two other hazards that may impede a computer system. (4marks)

(b) Explain three reasons why wireless communication is not preferred by some organizations. (3marks)

(iii) Explain the consequences of network communication without each of the following protocols (relate your answer to the purpose of the protocol).

(a) HTTP.
(b) FTP.
(c) TCP/IP. (3marks)

(iv) State two wireless standards and give an estimation of their area of coverage. (2marks)

---

5. (i) With regards to the relational database, discuss with the aid of diagrams, the meaning of the following:-

(a) One-to-one relationship. (2marks)
(b) One-to-many relationship. (2marks)
(c) Many-to-one relationship. (2marks)

(ii) With the help of well labeled diagrams, describe three commonly used standard network topologies, giving one advantage and one disadvantage for each case. (6marks)

(iii) Briefly describe each of the following.

(a) E-commerce.

(b) Browser.
(c) E-mail.
(d) Search engine.
(e) DNS. (5marks)

---

6. (i) List and explain TWO main threats to the safety of data. (4marks)
   (ii) How does the Data Protection define the following terms?
   (a) Personal data.
   (b) Data.
   (c) Data Subject. (3marks)
   (iii) Differentiate between "synchronous" and "asynchronous" data transmission. (2marks)
   (iv) Explain each of the following project management terminologies.
   (a) Project.
   (b) Critical path.
   (c) Gantt chart. (5marks)
   (v) Name three important factors to consider when designing a new Computer User Interface. (3marks)

---

7. (i) Briefly describe the following system software starting the category of each.
   (a) Compiler.
   (b) Defragmenter.
   (c) File compression.
   (d) Interpreter. (4marks)
   (ii) State FOUR functions of an operating system when managing a computer's resources. (4marks)
   (iii) Define the term interrupt. State one scenario in which an interrupt is used in the management of an operating system. (2marks)
   (iv) (a) What is a pseudo-code? (1mark)
   (b) Write a pseudo-code that request marks for 5 subjects and calculates the average assuming all subjects have equal coefficient. *Note that average is given by (total marks)/5 in case.* (6marks)

---

8. (i) Briefly describe the following systems as used in ICT, giving an example of each.
   (a) Monitoring system
   (b) Control system (6marks)
   (ii) With respect to computer usage, what is ergonomics? (1mark)
   (iii) Describe two ways that can be taken to prevent:
   (a) Carpal tunnel syndrome (CTS).
   (b) Eye strain. (4marks)
   (iv) Describe how transfer of funds is accomplished in E-commerce. (2marks)

---

9. (i) Define the following terms:

(a) Process
(b) Interrupt
(c) Deadlock (6marks)

(ii) List two major activities of an operating system with respect to:
(a) Memory management
(b) Secondary storage management (4marks)

(iii) Differentiate between multiprocessing and multiprogramming. (4marks)

(iv) (a) What is telemedicine? (1mark)
(b) Name TWO benefits of telemedicine to remote communities in Cameroon.(2marks)

GCEB JUNE 2016

1. (i) Explain the following computer terms giving examples where necessary:
(a) Video conferencing
(b) Stock control system
(c) E-learning (6marks)

(ii) (a) Explain one stage of the machine instruction cycle that takes place in the ALU of the CPU. (2marks)
(b) Sketch the logic circuit for the logic expression (***p*** *AND* ***q****) OR ()* (3marks)

(iii) (a) What is software reuse with respect to system development? (2marks)
(b) State two criteria used to select existing software for reuse. (2marks)
(c) Explain two situations that may cause organization to engage in externally developed (outsourcing) software. (2marks)

---

2. (i) (a) Explain the main difference between the peer-to-peer and the client/server networks. (3marks)
(b) Give two disadvantages of the client/server network. (2marks)

(ii) (a) What is an Information System? (2marks)
(b) Explain the term “procedure” as a component of an information system. (1mark)
(c) In terms of information system describes GIS and LIS. (4marks)

(iii) (a) What is the main difference between a high level language (HLL) and a low level language (LLL)? (2marks)
(b) Explain what is meant by machine dependent.
Which of these languages is machine dependent: HLL or LLL? (2marks)
(c) Give one example of HLL. (1mark)

---

3. (i) (a) What is SDLC? (1mark)
(b) Give two advantages of direct system implementation method. (2marks)
(c) State three activities involved in the design stage of SDLC. (3marks)

(ii) (a) What is the role of a system analyst in an organization? (2marks)
(b) Describe two ways by which a system analyst can get information about a system to be developed. (3marks)

(iii) Discuss with examples three changes in the workplace due to the increased use of ICT. Your answer should refer to the changes for the employees. (6marks)

---

4. (i) (a) What is an operating system? (2marks)
   (b) Explain the role played by the operating system in the given instances
   - Integrating additional hardware to a computer system
   - Managing RAM usage during processing (4marks)

   (ii) (a) What is a computer interface? (1mark)
   (b) Give the characteristics of the Graphical User Interface. (2marks)
   (c) State two advantages of GUI over Command Line. (2marks)

   (iii) Explain the effect and the consequences on computer system in the absence of each of the following:
   (a) RAM
   (b) Buffers
   (c) NIC (3marks)

   (iv) (a) What is memory? (1mark)
   (b) Describe the purpose of ROM in the computer system. (2marks)

---

5. (i) Structural programming refers to a general methodology of writing good programs.
   (a) What is programming? (1mark)
   (b) Give 4 properties of a good program. (4marks)

   (ii) Describe briefly the following :
   (a) Application software (2marks)
   (b) Virtual reality (2marks)
   (c) Biometrics (2marks)

   (iii) Give three advantages of simulation (3marks)

   (iv) Write the full meaning of each of the following abbreviations?
   (a) ASCII (1mark)
   (b) PAN network (1mark)
   (c) EPROM (1mark)

---

6. (i) It is usually necessary to compare different implementations of an algorithm to choose an optimal one. This can be done by considering their relative efficiencies.
   (a) What is efficiency with regards to algorithm performance? (1mark)
   (b) Which two basic resources of a computer can be used to measure the efficiency of an algorithm? (2marks)

   (ii) (a) Describe simulation and a situation that can be simulated. (3marks)
   (b) Give three advantages of simulation to architects. (3marks)

   (iii) Work generally could be classified into operations or project. Although in some cases, they do overlap.
   (a) Give any two characteristics common to operations and project. (2marks)
   (b) Make a difference between operations and project. (2marks)

   (iv) In a multiprogramming and time-sharing environment, several users share the same system simultaneously. This situation can result to various security problems.
   (a) What are two such security problems? (2marks)

(b) Is it possible to ensure the same degree of security in a time-share machine as in a dedicated machine? Explain your answer. (2marks)

---

7. (i) During the late 1960s and 1970s, there had been an outcry over impending "software crises". The symptoms of such software crises still exist in present day. List 4 symptoms of software crises. (4marks)

(ii) Explain the following terms:
(a) Parity bit (2marks)
(b) ARPANET (2marks)
(c) Internet standard (2marks)

(iii) (a) What is prototyping? (2marks)
(b) List **three** benefits of prototyping. (3marks)

(iv) Given that a disk has 18 sectors, 80 tracks and a sector capacity of 512bytes. Prove that the storage capacity is 1.44MB. (2marks)

---

8. (i) Explain briefly the meaning of each of the following terms giving examples where necessary.

(a) Automatic Speech Recognition (ASR) (3marks)
(b) HTML (2marks)
(c) Artificial Intelligence (3marks)
(d) Extranet (3marks)
(e) MIS (3marks)

(ii) (a) Explain the meaning of the term "Data Security". (1mark)
(b) Describe how encryption will help to protect information or a message which is sent across a network. (2marks)

---

9. (i) Define the following terms of online Internet services giving TWO advantages and ONE disadvantage of each;

(a) E-Commerce (3marks)
(b) E-Banking (3marks)
(c) E-Health (3marks)

(ii) (a) What is a social network? (2marks)
(b) Outline TWO advantages and ONE limitation of social networking. (3marks)

(iii) Explain the following errors that can arise during programming:

(a) syntax errors
(b) run-time errors
(c) logic errors (3marks)

GCEB JUNE 2017

1. (i) Briefly describe the following systems as used in ICT, giving an example of each.

(a) Monitoring system
(b) Control system (6marks)

(ii) The manager of a local plantation company complains that the management information system of the company is failing to provide the correct level of information.

(a) What is management information system? (2marks)
(b) Give four possible reasons why the system is failing. (4marks)

(iii) (a) What is digital divide? (2marks)
(b) Describe three causes of digital divide with respect to global communication. (3marks)

---

2. (i) What is ergonomics with respect to computer usage? (1mark)

(ii) Describe two ways that can be taken to prevent,

(a) Repetitive Strain Injury (RSI)
(b) Eye strain (4marks)

(iii) Distinguish between the following data transmission modes, giving an example of each.

(a) Serial and parallel transmissions.
(b) Broadband and baseband transmissions.
(c) Synchronous and asynchronous transmissions. (6marks)

(iv) Describe the following terms as used in Internet communication:

(a) Chat room.
(b) Instant messaging.
(c) Web log (blog). (6marks)

---

3. (i) (a) Differentiate between data and information. (2marks)

(b) Outline three characteristics that make data reasonable. (3marks)

(c) Rearrange the following to form the Data Hierarchy System starting with the smallest unit of data to the biggest. Fields, Data warehouse, Folders, Bits, Records, Files, Bytes, Word. (4marks)

(ii) (a) Briefly explain the term prototyping as used in software development. (2marks)

(b) Outline four stages involved in prototyping. (4marks)

(c) State two benefits of prototyping to the end user. (2marks)

---

4. (i) An organization has decided to connect its computers to communicate in a network.

(a) Give four hardware resources that will be needed in order to have Internet connection. (4marks)

(b) Describe two responsibilities of a network administrator in this organization. 2marks)

(ii) (a) Define the programming terms; syntax and semantics. (2marks)

(b) Describe each of the following: Algorithm, pseudo code and program. (3marks)

(c) Draw a flowchart to represent the following algorithm

Start,
Enter the radius, R
Compute Area, $A = \pi * r * r$
Print "A" and "R",
Stop. (4marks)

(iii) Give two advantages of data normalization. (2marks)

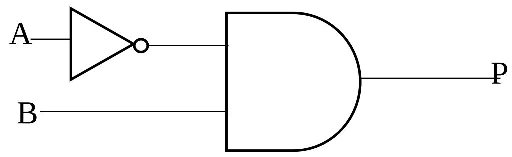
---

5. (i) What is computer crime? (2marks)

(ii) State two weaknesses of a bank's information system that may expose it to each of the following:-

(a) Unauthorized access to confidential data.
(b) Damage of data. (6marks)

(iii) Study the logic circuit diagram below:



(a) What is the value of P? (2marks)
(b) Draw the truth table of the logic circuit above in (a). (2marks)
(c) Draw the circuit diagram for a NOR gate. (2marks)

(iv) Identify and describe three criteria to be used for the selection of library software. (3marks)

---

6. (i) (a) Discuss **two** types of information systems that are useful to top level managers of an organization. (4marks)
(b) Name and explain **two** criteria used to judge the success of an information system in an organization. (4marks)
(ii) (a) What is a project? (2marks)
(b) Give **three** reasons why regular reviews are necessary during all the phases of a project. (3marks)
(iii) Define the following giving one advantage in using them to control a project:
(a) Critical path
(b) Flow chart (4marks)

---

7. (i)(a) In which phase of the SDLC is feasibility studies carried out? (1mark)
(b) Name and explain three types of feasibility assessments carried out in the phase. (3marks)
(c) State and explain three major tasks carried out during the design phase of the SDLC. (3marks)
(ii) What is e-commerce? (3marks)
(iii) Describe how the following activities are accomplished in e-commerce.
(a) Purchasing

(b) Selling
(c) Transfer of funds. (6marks)

(iv) Describe the Boolean data type and explain a situation in which it is used. (2marks)

---

8. (i) (a) What is the importance of file format? (1mark)
(b) Draw the table below and classify the following file formats in the right column of file type:

.JPEG, .MDB, .PDF, .XLS, .XML, .WAV, .PNG

Sound	Document	Spreadsheet	Bit-mapped		Database	Hypermedia

(4marks)

(ii) Describe four ways in which e-government could enhance government services in Cameroon. (4marks)

(iii) Describe and give an example of each of the following ICT systems. Your example may be a description of where the system is applicable.
(a) Expert system.
(b) Embedded systems. (4marks)

(iv) Name and explain two data transfer checks use in data communication. (4marks)

---

9. (i) (a) List and explain any two types of wireless network standards (4marks)
(b) Explain two disadvantages of wireless network. (2marks)
(c) Discuss how signal strength can be amplified in wireless networks. (2marks)

(ii) (a) Distinguish between the Internet and the Web. (2marks)
(b) Give and explain four ways in which the Internet has affected the society. (4marks)

(iii) If a novel contains 2500 characters how much storage space is required to contain an e-library with two million novels? Give your answer in gigabytes. (3marks)

BIBLIOGRAPHY

[1] Adewoyin, J.A (2004). "*Fundamentals of Information Technology*", Lagos: Rayel Communications Limited.

[2] Alan Evans, Kendal Martin, and Mary Anne Poatsy, "*Technology in action*", 8th Edition, Pearson Education.

[3] B. Ram, "*Computer Fundamentals*", New Age International Pvt. Ltd

[4] Butler, Jeremy G., *A History of Information Technology and systems, University of Arizona, 2012*

[5] Edge, David. "*The social shaping of Technology*". Information technology and society: A reader. London: Sage Publications in association with the Open University. (1995)

[6] Garry J. Marshall. "*Beginner's guide to Information Technology". Michigan: Newnes Technical Books (2007)

[7] Heinich, R., Michael, M., James, D.R., Sharon, ES (2002). "*Instructional Media and Technologies for Learning", (7th Edition). Ohio, Pearson Education, Inc.

[8] Hundley, Richard…[et al]. "*The Global course of the Information Revolution: Recruiting Themes and Regional Variations.*" Saint Monica, CA: RAND, 2003.

[9] Kumar, K.L. "*Information Technology, Principles and Application"*, New Age International. New Delhi. (2005)

[10] Martin, William J. "*The Global Information Society*". UK: Aslib Gower. (1995)

[11] Norton, Peter, "*Introduction to Computers*", Mc-Graw-Hill.

[12] Raja Raman V., "*Fundamental of Computers*" (4th Edition.), New Delhi. Prentice Hall of India.

[13] S. Jaiswal, "*Fundamental of Computer & IT*", Wiley dreamtech India.

[14] Trainer T., et al, "*Computers*", McGraw Hill.

[15] Webster, Franck. (2003). "*Theories of the Information Society*", 2nd Edition London: Routledge.

[16] Webster, Frank. (2004). *the Information Society Reader*. London: Routledge.

[17] Yongho S. Louis, "*concepts in Computer Science & Information Technology*", University of Bamenda, 2015.

[18] Torben Lage Franden and bookboon.com, "*Excel 2007*", ISBN 978-87-7681-675-9, 2010, Ventus Publishing ApS.

[19] Thomas Malone, Chris Dellarocas, Stuart Madrick. 2005. *Lecture Notes Inforamtion Technology*. [ONLINE] Available at: http://ocw.mit.edu/courses/sloan-school-of-management/15-561-information-technology-essentials-spring-2005/lecture-notes/.

[20] BBC. 2016. *BBC Bitesize-GCSE Computer Science*. [ONLINE] Available at: http://www.bbc.co.uk/education/subjects/z34k7ty.

[21] BBC. 2016. *BBC Bitesize-GCSE ICT*. [ONLINE] Available at: http://www.bbc.co.uk/education/subjects/zqmtsbk.

[22] Tutorialspoint. 2016. *Tutorials for OAuth2.0*. [ONLINE] Available at: http://www.tutorialspoint.com/.

[23] Tutor2u. 2015. *What is ICT?-GCSE,* [ONLINE] Available at: http://www.tutor2u.net/business/reference/what-is-ict